昭和58年11月1日発行（毎月1回1日発行）第2巻第11号通巻第18号　昭和58年6月7日国鉄首都特別扱承認雑誌第6897号

パソコン情報誌　Personal Computer Magazine

480円

Oh!MZ

MZ, X1&ポケコンシリーズ

特集
ユーティリティプログラム
BASICユーティリティ/カセットコントロール/パターンエディタ/漢字作成ツール/マシン語入力ツール

X1シリーズ登場
速報 ニュータイプX1C, X1D

入門F-DOS/CP/M

MZ-LOGO音楽演奏

カラーUFOゲーム

マシン語ゲームMONKEY THREE

SPACE FIGHTER

X1マシン語入門
X1 システムソフトの活用

ゲーム作成入門講座
X1 THE PRO-BOWLING

ハード講座
X1強化実践教室



11 NOV.1983

SHARP

3つの新開発で実現



した「使いこなせる16ビット」

「16ビットにしたいが生かしきれないのでは………」
こうした声に、いまシャープがお応えします。
従来のコンセプトを一新し、
より人間の世界に近づけた汎用16ビット、MZ-5500シリーズの登場です。

マルチウインドウ

《複数画面(データ)を一度に表示》最大4画面をCRTで一覧できます。しかもCRT上の任意の位置に任意の大きさで表示でき、優先順位つきで重ね合わせも可能。いわば4ページにわたるデータを並べて判断できるわけで、ビジネス能力を飛躍的に高めます。またハードウェアで実現したことにより、ソフトウェア処理では不可能だった高速画面処理も可能になりました。

マウス※1

《動かすだけの簡単入力》話題の座標入力装置を採用しました。画面に集中しながら、手を動かすだけで、誰にでも簡単に操作できます。項目選択を行うビジネスソフトに、グラフィックスに、複雑なキー操作から解放されたかってないユーザーフレンドリーなソフト展開が図れます。

ビットマップ表示

《見やすいドット単位のスクロール》ディスプレイ上の1ドットがビデオRAM上の1ビットに対応する完全グラフィックディスプレイを実現。キャラクタとグラフィックの混在したハードコピーが容易にとれる他、リスト表示などスクロールの際、文字が非常に読みやすくなっています。

■CP/M-86™※2、漢字CP/M-86™※2(オプション・発売予定)、MS-DOS™※3(オプション・発売予定)など充実したソフトウェア環境■MZ-2000系BASIC採用、さらにGW-BASIC™※3(オプション・発売予定)もサポート■本格16ビットCPU8086搭載■高速数値演算プロセッサ8087(MZ-1M03・オプション・発売予定)サポート※4■最大192Kバイト(96Kバイト標準装備)の大容量ビデオRAMでモノクロ時最大8階調2画面、カラー時最大8色2画面をコントロール■カラーパレット機能、カラープライオリティ機能など多彩なグラフィック機能■JIS第一水準の漢字メモリ(MZ-1R10・オプション)の他、辞書ROMボード※4(MZ-1R14・オプション・発売予定)のサポートでワープロレベルの日本語処理■最大512Kバイトのメインメモリ■320Kバイトのミニフロッピー内蔵(MZ-5501を除く)■8オクターブ3重和音のサウンドジェネレータ■カレンダークロック機能■新JIS配列、シリンドリカルステップスカルプチャータイプの使いやすいキーボード■RS-232C(2チャンネル)、プリンタ(セントロニクス準拠)など各種インターフェイス標準装備■10Mバイトのハードディスク(オプション・発売予定)をサポート

※1 オプション　※2 CP/M-86™、漢字CP/M-86™は米デジタルリサーチ社の登録商標です。　※3 MS-DOS™、GW-BASIC™は米マイクロソフト社の登録商標です。　※4 発売予定の漢字CP/M-86™にてサポート。



16ビットパーソナルコンピュータ

MZ-5500シリーズ

MZ-5521	標準価格388,000円	●256KバイトRAM標準装備・ミニフロッピー2基内蔵
MZ-5511	標準価格288,000円	●128KバイトRAM標準装備・ミニフロッピー1基内蔵
MZ-5501*	標準価格218,000円	●128KバイトRAM標準装備

※MZ-5501は、従来のMZシリーズ用ミニフロッピーMZ-1F07、MZ-1F02、MZ-80BFをお持ちの方のためのモデルです。

◀写真はMZ-5521。12型カラーディスプレイMZ-1D11(標準価格113,000円)、チルトスタンドMZ-1S08G(標準価格10,000円)およびマウスMZ-1X10(標準価格19,800円)はオプションです。また画面はオプションの漢字メモリMZ-1R10(標準価格30,000円)を装着した例で、ハメコミ合成写真です。

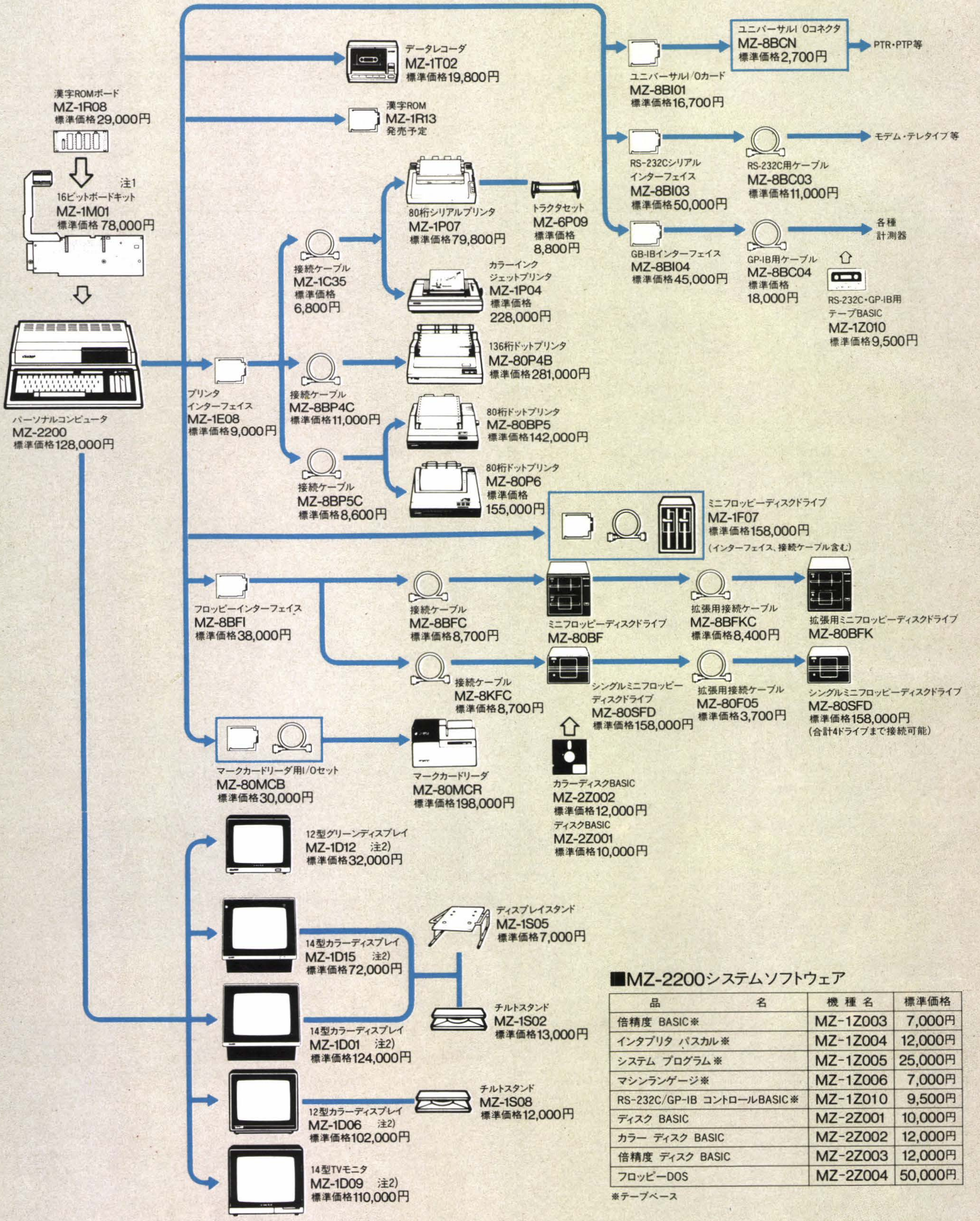
シャープ株式会社 本社 〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221(大代表)
●お問い合わせ、資料請求は…シャープ㈱国内産機営業本部 〒162 東京都新宿区市谷八幡町8番地 ☎(03)260-1161(大代表)
●シャープパーソナルコンピュータMZシリーズには、16ビットのMZ-5500シリーズの他、8ビットのMZ-3500シリーズ、MZ-2200、MZ-700シリーズがあります。

資料請求券
MZ-5500
Oh!!MZ 11月号

SHARP

充実したサポート体制で幅広い拡張性を誇る

●MZ-2200システム構成図



■MZ-2200システムソフトウェア

品　名	機 種 名	標準価格
倍精度 BASIC※	MZ-1Z003	7,000円
インタプリタ パスカル※	MZ-1Z004	12,000円
システム プログラム※	MZ-1Z005	25,000円
マシンランゲージ※	MZ-1Z006	7,000円
RS-232C/GP-IB コントロールBASIC※	MZ-1Z010	9,500円
ディスク BASIC	MZ-2Z001	10,000円
カラー ディスク BASIC	MZ-2Z002	12,000円
倍精度 ディスク BASIC	MZ-2Z003	12,000円
フロッピーDOS	MZ-2Z004	50,000円

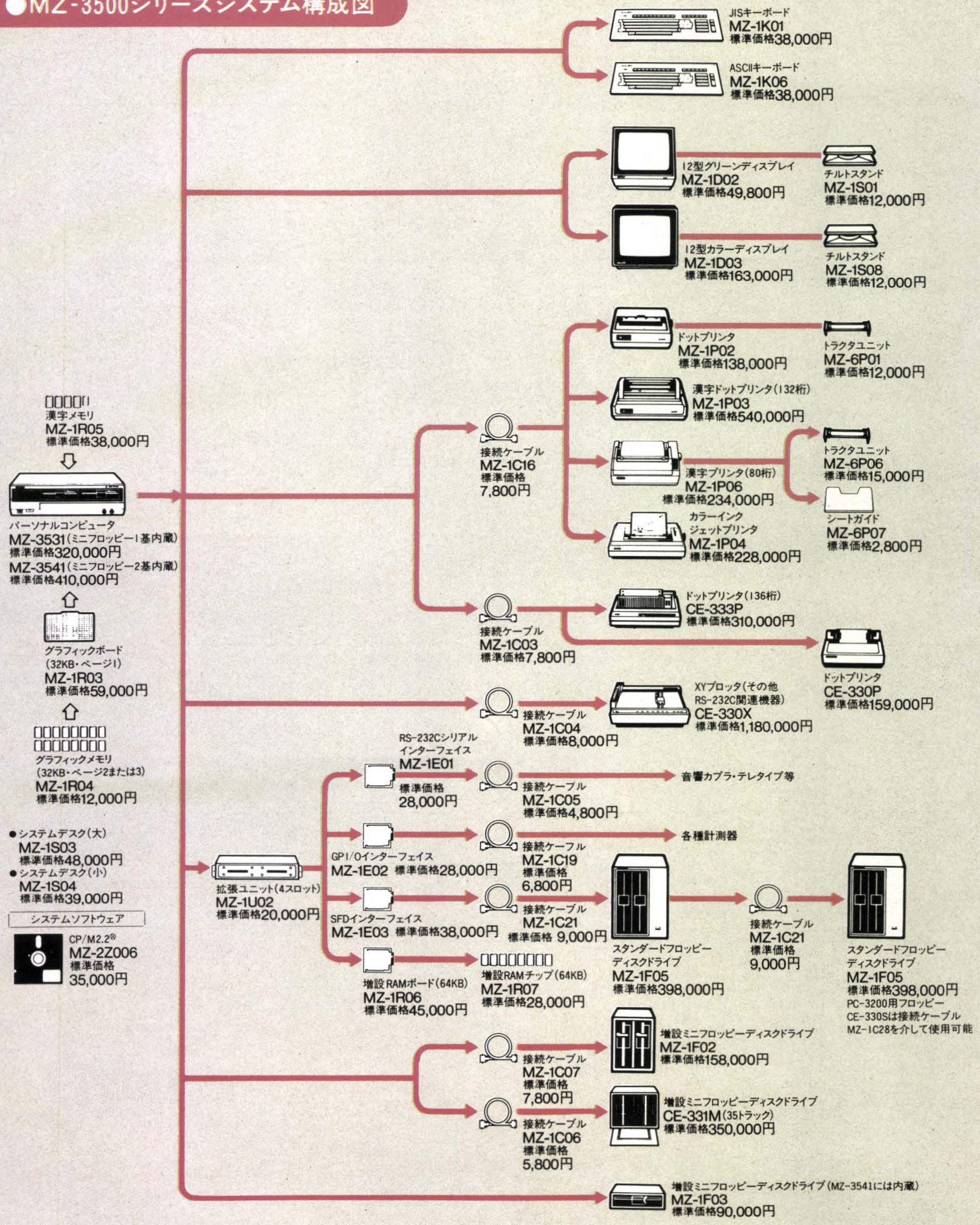
※テープベース

注1) 16ビットボードキット MZ-1M01を使用するには専用データレコーダ MZ-1T02(オプション)が必要です。

注2) MZ専用ディスプレイには、MZ-2200との接続ケーブルが付属しています。

MZクリーンコンピュータ。

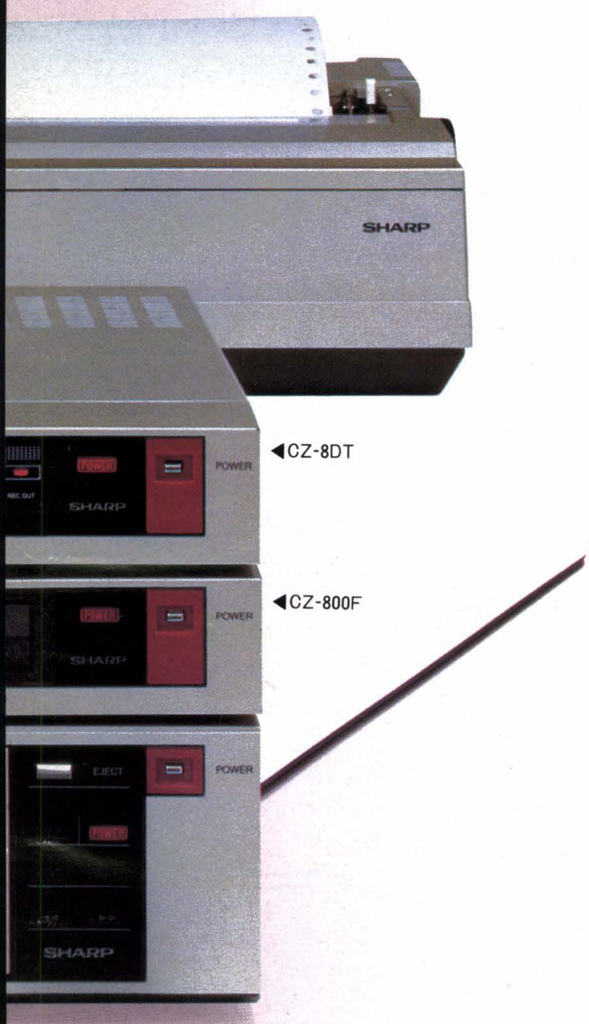
●MZ-3500シリーズシステム構成図



SHARP

グラフィック最先端の興奮を。





●シャープのエレクトロニクス技術が、パソコンとテレビの独創的なシステムを実現。パソコンの機能がテレビの機能を拡大し、テレビの能力がパソコンの能力を一気に高めました。いまやパソコンはグラフィックアビリティで選ばれる時代、映像表現に新領域をひらいたスーパーインポーズ機能をはじめとしたX1の新鮮なグラフィックスは、まさにプロのテクニックに迫る充実ぶり。ビギナーからエキスパートまで多くの人々を魅了しています。●もちろん、パソコンとしての高機能も見逃がせない特長です。3CPU構成、伝統のクリーン思想を受け継いだメインメモリ64Kバイト、オールRAM。自動頭出し機能を備えた2700ボーの高速カセットレコーダ内蔵。そしてこうしたハードウェアに密接に結びついてその機能を最大限に発揮させる強力なソフトウェア。X1は、究極の8ビットマシンとして、ホームユースからビジネスユースまで幅広い分野への応用が可能な、まさに21世紀のメディアです。

●メインCPUにZ80A(4MHz)、周辺コントロール用として2個のサブCPU搭載●RAM：メインメモリ64Kバイト、テキスト用V-RAM4Kバイト、ユーザー定義キャラクターゼネレータ6Kバイト、グラフィック用V-RAM(48Kバイト・オプション)●カラーBASICインタプリタ装備●セントロニクスプリンタインターフェイス、ジョイスティックインターフェイス内蔵●8オクターブ3和音のサウンドゼネレータ●日付、時刻、番組予約が可能なカレンダー付タイマー●小さな文字や細かな縞模様もくっきりと再現する新開発ファインピッチブラウン管●RGB接続端子とビデオ入出力端子付

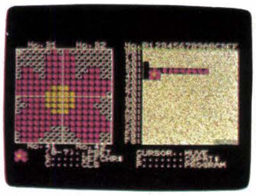


●映像表現の新領域――スーパーインポーズ機能

X1は従来むずかしいとされていたコンピュータ画像とカラーテレビの画像を、同じブラウン管上で重ね合わせること(スーパーインポーズ)に成功。新開発のASC(オートマチック・シンクロナイズ・コントロール)回路とファインピッチブラウン管の採用により実現したものです。さらにビデオやビデオディスクと接続して、その画像とコンピュータ画像を重ね合わせることも可能。スーパーインポーズされた画像は、デジタルテロッパー(別売)の使用により、ビデオ録画もできます。

主な拡張用オプション

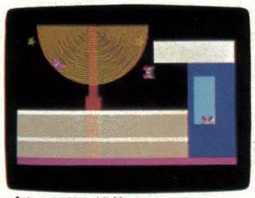
フロッピーディスク	CZ-800F	標準価格198,000円
フロッピーディスクインターフェイス	CZ-8FA	標準価格 24,000円
ドットプリンタ	CZ-800P	標準価格142,800円
デジタルテロッパー	CZ-8DT	標準価格 89,800円
グラフィックRAM	CZ-8GR	標準価格 32,000円
漢字ROM	CZ-8KR	標準価格 38,000円
拡張I/Oポート	CZ-8EP	標準価格 11,800円
RFビデオコンバータ	CZ-8VC	標準価格 15,800円



キャラクターゼネレータ



パレット機能



プライオリティ機能

〈パソコンテレビX1が誇る強力なグラフィック機能〉●256種類のキャラクターが自由に定義できるプログラマブルキャラクターゼネレータ●多画面切換えができる高機能グラフィック●図形や文字の色を瞬時に変えられるパレット機能●遠近感、立体感のある画面がつくれるプライオリティ機能●グラフィックの拡大縮小、移動ができる座標変換機能●任意の面の色づけが簡単にできるタイリングペイント機能●大小の文字を同一画面に混在できるアトリビュート

パソコンテレビX1

パーソナルコンピュータ+キーボードCZ-800C 標準価格155,000円・14型カラーディスプレイテレビCZ-800D 標準価格113,000円

●写真はパーソナルコンピュータ+キーボード(CZ-800C)、14型カラーディスプレイテレビ(CZ-800D)、フロッピーディスク(CZ-800F)、デジタルテロッパー(CZ-8DT)、ドットプリンタ(CZ-800P)を組合せた例です。なおスーパーインポーズはCZ-800CとCZ-800Dのシステムで作成可能。●画面はオプションのグラフィックRAMを使用し作成、印刷合成したものです。

New Life Now

▶パソコンテレビX1には、ローズレッド・メタリックシルバー・スノーホワイトの3色があります。

シャープ株式会社 本社 〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221(大代表)
●お問い合わせ、資料請求は…本社内電子機器事業本部システム機器営業部
またはシャープエンジニアリング㈱本社 〒114 東京都北区東田端2丁目13番17号 ☎(03)893-4649
およびシャープビジネス㈱本社 〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221(大代表)へ。

資料請求券
X1
oh!!MZ
11係

S・CALGO

エスカルゴ

MZ-700用H・Aソフト(簡易表計算プログラム)

ホーム・オートメーション

知は力なり（ソフト／パワー）



好評発売中!!

テープ版¥9,800

家庭で、オフィスで、商店で、学校で、表集計作業はたいへん多いものですね。家庭では家計簿や住所録、オフィスや商店では売上集計表や在庫一覧・顧客名簿、そして学校では成績一覧表やスケジュール表等々。これまで、紙と鉛筆、電卓(そろばん)で行なっていた作業をMZ-700を使って手軽に、スピーディに行なうソフトが、このS・CALGOです。H・Aに、スモールビジネスに威力を発揮します。

特　長

- ●データ入力、演算、印刷等の操作が、ファンクションキー、画面メッセージにしたがって、簡単にできます。
- ●オールマシン語により、処理が高速です。
- ●用途に合わせて、2タイプの表が選択できます。
- ●ソート、検索などの機能も充実しています。

処理機能

S・CALGO

F1	DEFINE	(演算定義)
F2	JUMP	(指定行へのカーソル移動)
F3	IN/DEL	(行、列の挿入・削除)
F4	FILE	(表)
F5	EXEC	(演算実行)
F6	GETPUT	(項目のコピー、クリア)
F7	SEARCH	(検索)
F8	SR/LK	(ソート・ロック)
F9	PRINT	(印刷)
F10	OPTION	(宛名印刷、グラフ作成等のユーティリティプログラム・別売)

S・CALGOオプションパック新発売!!

テープ版¥9,800

"S・CALGOオプションパック"は、S・CALGOにさらに多彩な機能をもたせ、より活用範囲を広げるプログラムです(A面)。さらにB面には、事例データファイルをつけています。

機能・特長

- ●郵便の宛名印刷
- ●入力した文章をそのままプリンタに出力し、タイプライター的に使う機能。
- ●棒グラフ、円グラフ、折れ線グラフの作成(プロッタ・プリンター要)
- ●住所録、売上管理表の事例データファイルによって、S・CALGOの操作方法がより容易に理解できます。

CTC 株式会社 東海クリエイト

〒108 東京都港区芝5-13-18 MTCビル6F

ソフト営業部TEL 03-456-4615

インフォメーションセンター

(専用) TEL 03-456-4610

●当社ソフトについてのお問合せは上記までお気軽にどうぞ。


MagicalZoo
おもわず冒険！
グラフィックアドヴェンチャーゲーム
ムー大陸の謎
ピラミッドの謎
黄金の墓
○グラフィックアドヴェンチャーゲームシリーズ
カセット版、ディスク版（マニュアルつき）共に、
全国マイコンショップにて好評発売中！
5インチFD版 5,800円
カセット版 4,800円
FM-7/8
X1
（要V-RAM）
PC-8801
PC-6001mkII
/PC-6001
FOR X1
本格派グラフィック・アドヴェンチャーゲーム！
ちょっとやそっとじゃラストは見えない。コンパス
推理力・想像力をたよりに探険する。
これは君自身の、オデッセイ！
パソコンアニメーター
「ピラミッドの謎」の画面製作に使われた、
抜群のグラフィックツール！好評発売中
■同じパターンを繰り返し描くワープ機能つき ■グラフィックカーソルの座標をドット単位で表示、画面に方眼を表示グラフィックのモードも表示します ■どんな色でも自由に作れます（ディスク版のみ、カセット版は40色まで）■簡単な操作で直線・曲線・点・四角形・円・惰円が描けます ■絵の拡大、縮小、保存、呼び出し、プリントアウトも簡単です ■たった20のコマンドでどんな絵も自由自在 ■機能は充実、使いやすさは抜群■イラスト、デザイン、設計図から、創造性を延ばす幼児教育まで ■要V RAM
5インチFD版 9,800円
（ジョイスティック対応）
カセット版 5,800円
ひらがな百人一首
カセット版 4,500円
X1
JR-200
mz-
2200
2000
80B
マジカル ズゥ

SilverBall MZ-



Silver Ball（シルバーボウル）の「あそび」が MZ-2000/2200が 楽し

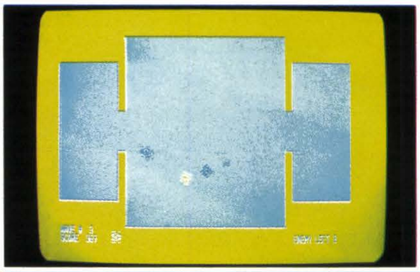
MZ-2000/2200 定価 各2800円◀



弾をさけての道路工事なんて。やってみるー？

mission Z2000
ミッション

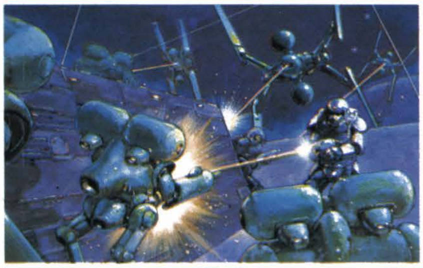
SHSK28001 発売中



相手の車を壁にぶつけ破壊し勝ち残れ！クラッシュレース

BUMPER CARS
バンパーカーズ

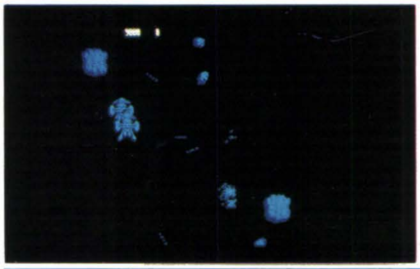
SHSK28006 新発売



スーパーマンになってみたかった。バキューン。

OMEGA TRON
オメガトロン

SHSK28002 発売中



接近するUFOや隕石をビーム砲で破壊するゲーム

UFO
ユーフォー

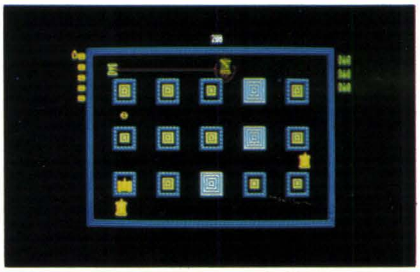
SHSK28007 新発売



戦時のパイロットの使命は確実に敵を破壊すること！

FLYBOY
フライボーイ

SHSK28003 発売中



戦車で敵地へ乗り込んだ君は、地雷を、爆破しながら敵戦車を攻撃しなければならない。

LASER TANK
レーザータンク

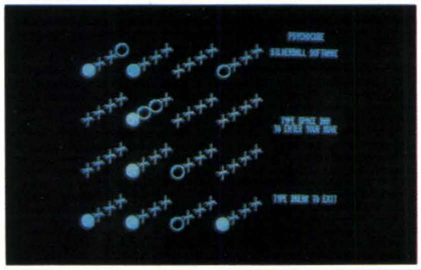
SHSK28008 新発売



この町には「胸キュン」がいる。ガンバラなくっちゃ。

DAWN PATROL
ドーンパトロール

SHSK28004 発売中



コンピュータとの知恵くらべ。3Dの4目並べです。

PSYCHO CUBU
サイコキューブ

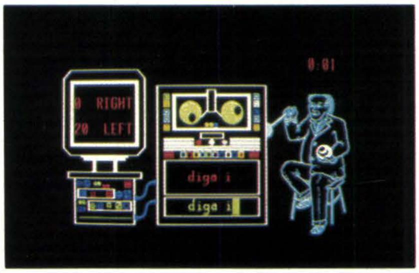
SHSK28009 新発売



エイリアンの変化の面白さが、戦いを忘れさせた。

UNDER GROUND CITY
アンダーグランドシティ

SHSK28005 発売中



天才アインシュタイン先生の操作によって、コンピュータが問題を出します。

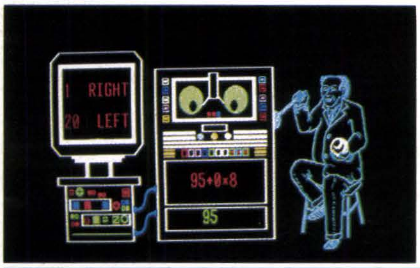
DR, QWERTY
ドクタークワァティ

SHSK28010 新発売

2000 2200 X1 USA版

きみの知能に挑戦する。めるアメリカン・ソフト新発売。

▶定価 各3000円



暗算のトレーニングで君の脳細胞を活性化しよう。学習用

DR, MATH
ドクターマス
新発売
SHSK28011



UFOの攻撃に反撃しながら宇宙船を惑星に着陸させよ！

DEATH PLANET BATTLE
デスプラネットバトル
新発売
SHSK28012

アメリカ生まれの面白ゲーム。
ピッと脳細胞に反応したソフトをRUNしてみよう。
スピードとリズムと「あそび」のゲームが
あなたの期待に応えてくれます。
カラーでPLAYすれば迫力バイバイゲーム。
ドクターシリーズは教育用プログラムです。

製造・発売元
SilverBall
シルバーボール ソフトウェア
2855, Telegraph Ave. Suite 216,
Berkeley, CA94705 U.S.A.
Phone (415)848-6666
Telex 338586 Unison Berkeley

お問い合せ：株式会社 日本ソフトバンク本社内
〒102 東京都千代田区九段南2-3-14
靖国九段南ビル2F Phone 03(263)3904

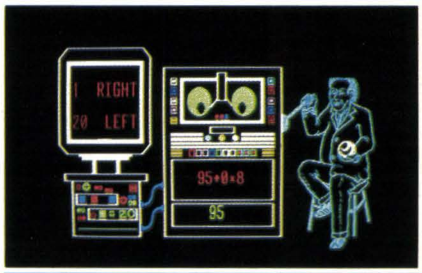
Silver Ball(商標)は、次々と新しいゲームを
発表・発売していきます。
ご期待ください。

X1 定価 各3000円



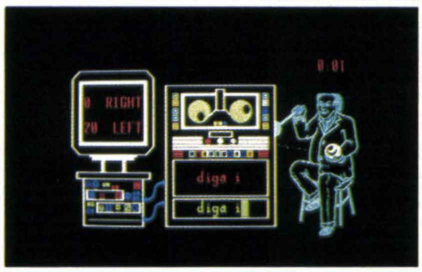
戦時のパイロットの使命は確実に敵を破壊すること！

FLYBOY
フライボーイ
発売予定



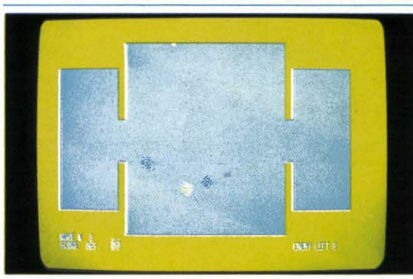
暗算のトレーニングで君の脳細胞を活性化しよう。学習用

DR, MATH
ドクターマス
発売予定



天才アインシュタイン先生の操作によって、コンピュータが問題を出します。

DR, QWERTY
ドクタークワァティ
発売予定



相手の車を壁にぶつけ破壊し勝ち残れ！クラッシュレース

BUMPER CARS
バンパーカーズ
発売予定



UFOの攻撃に反撃しながら宇宙船を惑星に着陸させよ！

DEATH PLANET BATTLE
デスプラネットバトル
発売予定

*このテープよりの複製及びレンタルを禁止します。

Hudson soft

※最新ソフトカタログをご希望の方はX1, PC-8001mkII, パソピア7, FM-7, 日立レベル3マークV, MZ-2200, PC-6001mkIIの機種名を明記の上ハドソン東京・OH-MZ係まで〒60円を添えて, お申し込み下さい。

※通信販売は, ソフト名を明記の上お近くのハドソン・OH-MZ係まで現金書留にてお送り下さい。（送料無料）

パソコンテレビX1用ゲームソフト

めだかのきょうだいがゲームになった!

© テレビ朝日ミュージック

X-1086

マシン語　¥3,000

めだか, こねこ, すずめのきょうだいが繰りひろげる楽しいメルヘンの世界!!



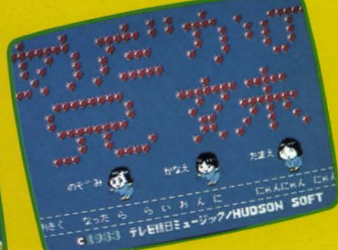
落ちてくるおにぎりを食べてね!
スイカを食べるとボーナス点です
スイカが上までいかないように。













落ちてくるブドウを食べてね!
サクランボはボーナス点です。

私に大好物のサカナを食べさせてね!
リンゴを食べちゃうとボーナス点ですよ。



※PC-8801, PC-8001mKII, FM-7, パソピア7, MZ-2200も開発中です。

HUDSON GROUP
HUDSON SOFT®
ハドソン札幌/〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸II 201 PHONE:011-821-1538
ハドソン仙台/〒980 宮城県仙台市宮町1丁目4番28号 PHONE:0222-65-7031
ハドソン金沢/〒920 石川県金沢市本町2丁目1番28号 PHONE:0762-23-1263
ハドソン東京/〒102 東京都千代田区麹町4丁目7番5号 麹町ロイヤルビル2F PHONE:03-234-4996
ハドソン大阪/〒542 大阪市南区南船場4丁目2番18号 佐野屋橋ビル5F PHONE:06-251-1945
ハドソンUSA/2063 CENTER STREET BERKELEY CA 94704 TELEPHONE 415-845-1416


STEP企画第1弾
Honeybee Soft
3びきの子ぶたの大冒険
夢がいっぱいの楽しい童話の世界へ入ってみませんか？
あなたは、STEP 1?・2?・3?
パソコンテレビX1,PC-6001mkII, PC-8801,MZ-2200用
ストーリーブック付
STEP 1
ここはお花畑。花のタネに水をやって花を咲せましょう。
花に水をあげると花の精になって飛んでいきます。
しかし、いたずら猿や猫が出てくるので水をかけてこらしめてください。
STEP 2
ここは森の広場。家を建てるための丸太をあつめましょう。
いじわるな猿が丸太を盗んでいきますから注意してくださいネ！
サルにバナナを投げて追いはらってください。
STEP 3
けれども仲間は好物を持っていかないとついて来てくれませんよ。
子ぶたがうさぎに森の仲間を集めるように命令します。
人にぶつかると好物がどこかに飛んでいっちゃうから気をつけてネ！
3びきの子ぶたの大冒険
マシン語　各¥3,200
企画：ハニービーソフト
監修：深沢清澄
※この画面は、当社モニターで撮ったもので機種によって画面は異なります。

maxell

漫画のアイデアに大切な「連想」を、いつかパソコンとやってみたい。



生物としてパソコンをとらえたい。

秋竜山さんのお宅にあるパソコンは、目下のところ息子さんの直良君（中1）が使用中。いずれ秋竜山さんも始めたいそうですが、その時は直良君が先生になりそうです。「パソコンを使い出したら、パソコンを生物としてとらえて、個性的に飼育したいね。つまり、漫画のアイデアには“連想”が大切なんですが、人間の連想を超えた連想を、パソコンとともにしていきたいということなんです」。将来の夢をこのように語ってくれました。また、必要なときに必要なものをすぐ取り出せる、百科辞典的な使い方もしてみたいとか。その頃には、マクセルのフロッピーディスクが、秋竜山さんとそのパソコンの、良きパートナーとして活躍することでしょう。



秋 竜山さん ナンセンス漫画家の第一人者

60℃の高温に耐えるHRジャケット。

マクセルのフロッピーディスクは、世界に先がけてHR(Hjgh-Temperature Resistant)ジャケットを本格採用しました。このHRジャケットは、60℃の高温下でも変形しない耐熱性を確保。もちろん、ディスクそのものもマクセルだけの、全天候型磁気ディスクを採用。こうしたマクセルの先進技術は、コンパクト・フロッピーディスクや、パーソナル・コンピュータカセットにも生かされています。



maxell FLOPPY DISK

マクセル「'84漫画家カレンダー」プレゼント！

6人の人気漫画家(久里 洋二・岩本 久則・古谷 三敏・秋 竜山・ちば てつや・ジョージ 秋山)が、マクセル・フロッピーディスクのため描いてくれた漫画を来年度のカレンダーにしました。これを抽選で2,000名様にプレゼント。どんどん応募して、部屋に貼

●応募方法：官製ハガキに郵便番号、住所、氏名、年齢、性別、職業、パソコンの有無(お持ちの方はフロッピーのドライブ装置の有をご記入の上お送りください。●応募期間：昭和58年11月1日～12月20日 ●応募先：〒104 東京都中央区銀座3-3-1日立マクセル会社・宣伝グループ「マクセル・'84漫画家カレンダー・プレゼントF・OM係」● 当選発表：賞品の発送をもって発表にかえさせていただき

日立マクセル株式会社　営業本部／東京都中央区銀座3-3-1〒104 ☎03-567-6221(代)●資料のご請求は宣伝グループF・OM

絶賛発売中

MZ-LOGO

日本ではじめてMZ-2000に搭載！

●MZ LOGOの特徴

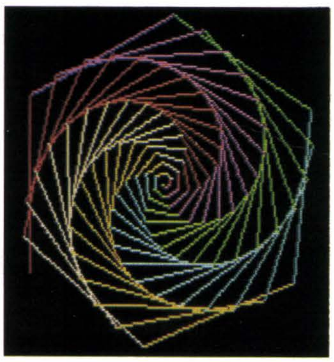
①パソコンで使えるプログラミング用高水準言語で，BASICと同じ会話型インタプリタです。
②"タートルグラフィック"というグラフィック機能を持っています。
③プログラムがプロシジャ構造（細かい手順に分けられた構造）になっていて，見やすくなっています。
④同じ作業を繰り返す再帰機能をもち，またユーザーがコマンドを自由に定義できます。
⑤リスト処理機能で文字も数字も同様に扱えます。
⑥作図，作画，ゲーム，人工知能，データファイルなど応用範囲が広く，初心者から専門家まで使えます。
⑦日本ではじめての和文LOGOマニュアル付（約100ページ）

●定価 ¥9,800（カセット版）

タートルグラフィックス　プログラム例

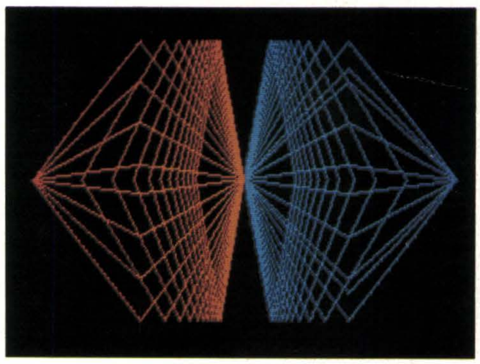
```
to egaku1
setpc 2
at -50 (- 50)
polyspi 100 62
end

to polyspi :side :angle
if :side < 2 [stop]
setpc pc + 1
if ((remainder pc 8) = 0) [setpc 2]
fd :side
rt :angle
polyspi :side - 1 :angle
end
```



```
to egaku2
home
rt 45
hishigata 15 0.1
lt 180
hishigata 15 0.1
setscrunch 1
end

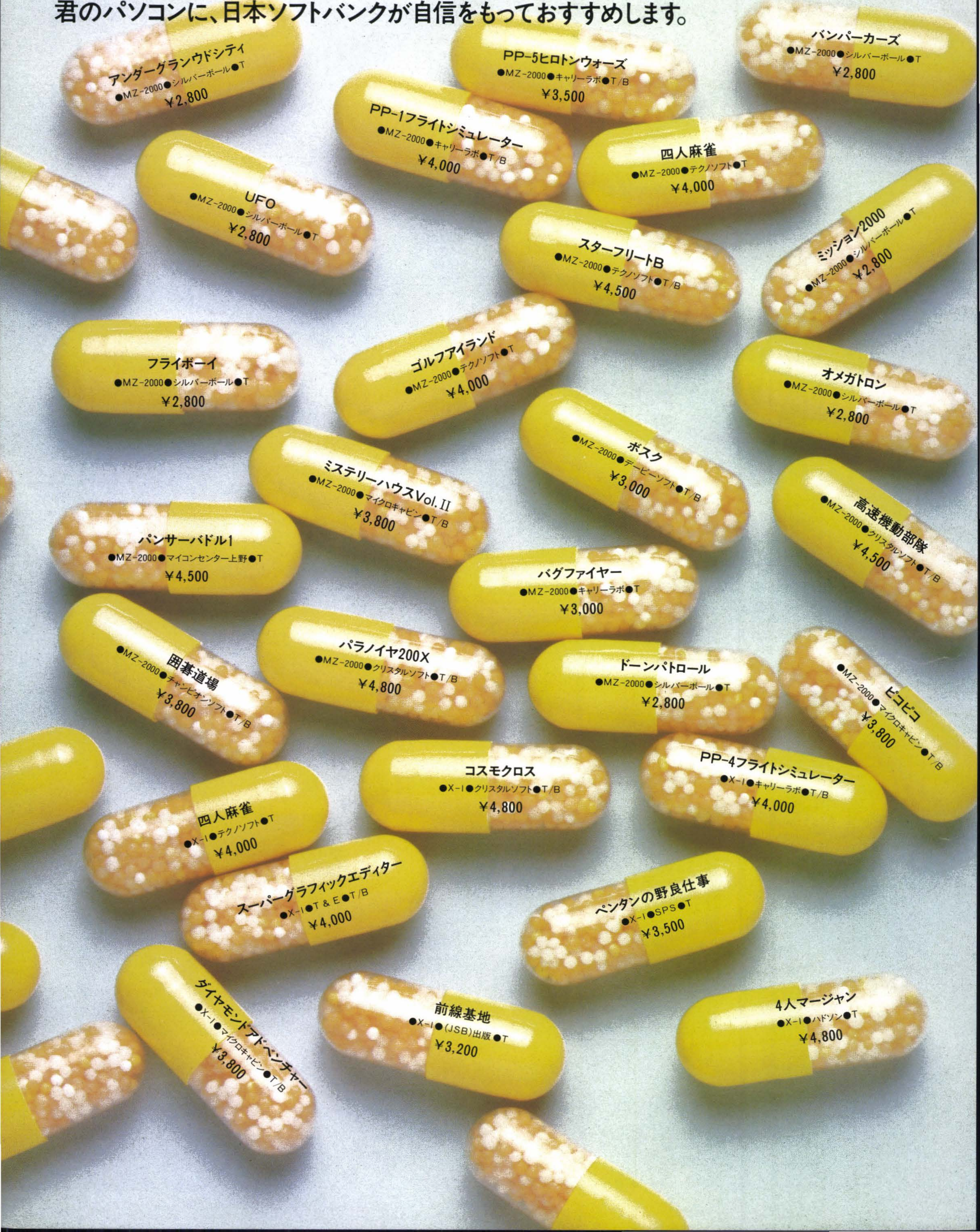
to hishigata :n :s
if :n = 0 [stop]
repeat 4 [fd 100 rt 90]
setscrunch :s
hishigata :n - 1 :s + 0.3
end
```



開発・発売元：UNISON WORLD INCORPORATED UW
2885 Telegraph Ave, Suite 216 Berkeley CA 94705 USA.
問い合わせ先：㈱日本ソフトバンクMZ LOGO係　〒102 東京都千代田区
九段南2－3－14 靖国九段南ビル TEL 03－263－3904

健康ソフト、ビタミンMZ。50種

君のパソコンに、日本ソフトバンクが自信をもっておすすめします。




ペンタンのアイスフィールド
●X-1●SPS●T
¥3,500
忍者くん
●X-1●マイクロキャビン●T/B
¥3,800
コスモファイター1
●X-1●日本ファルコム●T/B
¥3,500
探偵物語 2
●X-1●フィルコム●T/B
¥4,200
インペリアルフリート
●X-1●アポロテクニカ●T/B
¥3,300
スターフリートB
●X-1●テクノソフト●T/B
¥4,500
ゴルゴ13 PART 2
●X-1●フィルコム●T/B
¥4,800
4人マージャン(カラー版)
●MZ-700●テクノソフト●T
¥3,500
ドンキーゴリラ
●MZ-700●テクノソフト●T
¥3,500
パンサーパドル1
●MZ-700●マイコンセンター上野●T
¥4,500
パソコンアニメーター
●X-1●マジカルズゥ●T
¥5,800
聖なる剣
●MZ-700●クリスタルソフト●T/B
¥4,200
ワンダーハウス
●MZ-700●タスクフォーツ高知●T
¥3,500
宝ビルアドベンチャー2
●MZ-1200●マイクロキャビン●T
¥2,800
ギャラオ
●MZ-1200●富士音響●T
¥3,000
スーパーペンギン
●MZ-1200●富士音響●T
¥2,500
new VIP
●MZ-2000●デービーソフト●T/B
¥12,800
スーパーアセロ
●MZ-1200●九十九電機●T
¥1,500
Hu-ワープロ漢字
●X-1●ハドソン●ROM
¥24,800
グラフ 8
●MZ-700●パーソナルメディア●T/B
¥12,000
MZ-LOGO
●MZ-2000●ユニソンワールド●T/B
¥9,800
Hu-BASIC(Ver. II)
●MZ-2000●ハドソン●T/B
¥10,000

ファン待望のニューフェイス



X1C(CZ-801C)とチルトスタンドに乗せてあるディスプレイテレビ(CZ-801D) とのシステム例。とりあえずこの組み合せで従来のX1と同等の機能が使えます。写真ではシルバーですが，この他ローズレッドのタイプもあります。

X1C



X1C本体。キーボードはFM-7のキータッチを若干重くしたような感触でクリックが軽快。配列はX1と同じですね。キーボードの上右側がフルロジックカセットデッキ。その左側，写真では何もありませんがここにカラープロッタプリンタが入ります。



X1Cの本体裏側，各種拡張用のコネクタ類がならんでいます。左からRGB，テレビコントロール(これがないとX1じゃない)，拡張I/Oボックス用バスラインそしてプリンタ。右側にある2個の同じ端子はジョイスティック接続用です。



拡張I/Oボックス (CZ-8EB)。X1Cにフロッピーディスクを付けるためのインターフェースボードや漢字ROMボードなどを一度に最高4枚まで収納できます。



本体同梱の付属品。取扱説明書とHuBASICのマニュアル，そしてクリーンコンピュータでは当然のBASICテープ(HuBASIC CZ-8CB01)とデモプログラムの入ったアプリケーションテープ。

X1D



X1Dの基本システム。横幅ををオーディオラックにマウントできるサイズに統一してあるそうです。家電のシャープですからデザインも最高。色はローズレッドとシルバーメタリックの2種類があります。



キーボード部。従来のX1とほぼ同じですが，カセットがなくなりコンパクトフロッピーになったため，その制御キー3個がなくなりました。本体とのジョイントはまたしてもウォークマンタイプのステレオミニプラグです。

「やったね！」確か先月号で，MZシリーズの16ビットパソコンMZ-5500のとき，読者の方が編集部あてに寄せられたことばでした。MZの機種が増えちゃったものだから，ひょっとしてX1関係の記事が削られやしないかなあと内心ヒヤヒヤだった。でも，先月末からもうルンルン。なぜかわかります？　そう，ボクに仲間ができたんです。それも一度にふたつ。編集部の電話も連日新しいX1の問い合せがひっきりなし。これこそ「やったね！」と心からさけんでしまった。

Oh！MZは，ボクがここまでくるのにいろいろとお世話になっているから，そして先月号はとじ込みだけどボクの特集Oh！CZなんていうのもやってくれたし，本誌を毎月読んでくれている皆さんに，ボクの仲間をドーンと紹介しようと思うんだ。このページの写真で2つのモデルが写っているけど，これがニューフェイスたち。ボクは正式にはCZ-800Cというけど，愛称X1のほうが親しみやすいと良く耳にするので，彼らも愛称で紹介します。まずX1C，そしてX1D。どうぞよろしく。

どう？　なかなかカッコいいでしょ。個性たっぷりで。X1Cは，MZ-700やMZ-2200と同様，本体とキーボードが合体したんだ。でも良く見て！　データレコーダに操作ボタンなんかないもんね。そう，すべてソフトコントロールのフルロジックデッキ。左側には4色のカラープロッタプリンタだって付くんだ。ソフトも好き嫌いなんかせず，なんでも一発で読んじゃう信頼性バツグンのデッキ。それも1200ボーから2700ボーと2倍以上も早いんだ。他社のパソコンなんかに負けないゾ。そしてもうひとつのX1D。写真をよく見てくれればわかると思うけど，CPUユニット（本体）の右側に何かあるでしょ。これがすごいんだ。国産のパソコンでは初めて，3インチのコンパクトフロッピーを搭載してるんだ。小さいからといってバカにしたもんじゃない。なんと，ボクにオプションで付く5½インチのミニフロッピーと同じ320Kバイトも容量を持っている。初めは1基しか付いてこないけれど，必要ならばもう1基すぐ右側に増設できる。ね，すごいと思わない？

これがボクらX1シリーズのラインナップ。ボクはすごっくプライドが高い



CPU部前面。X1のロゴマーク上に2個あるがの新開発3インチコンパクトフロッピードライブ。写真では，1基増設しています。なかなか宇宙的なデザインですね。



X1D本体を裏側からみたところ。左側上の丸いものに注目。コンパクトフロッピーを内蔵したため，内部を冷却するファンが付きました。あとは，従来のX1とほぼ同じコネクタ類。一番大きなコネクタは拡張用フロッピー接続用です。



ワープロを始め，ビジネス用途に偉力を発揮する漢字プリンタ。16×16ドットで，漢字を出力します。



3インチコンパクトフロッピーディスクのメディアです。写真はシャープが同時に発売するブランクディスケットです。これにHuBASICを書き込んだものが本体購入時に付いてきます。

から，そうすべての人に使ってもらえるようなものじゃなかったけど，皆さんがもっと入手しやすいタイプをと，となえたのが通じたんだね。確かにボクを可愛がってくれている人の6割近くが40歳に満たない若い人だというし，そのうちの3割は大学生以下なんだって。欲しくても先立つものが……という人がけっこういることがわかったんだよね。X1Cは，119,800円。これなら手の届かない金額じゃないもの。で，X1Dはやっぱしちょっと高く，198,000円。でも，ボク用のミニフロッピー（CZ-800F）と同じ価格で，本体にコンパクトフロッピーが1基付いているんだ。どう，魅力でしょ。それも，普通のパソコンとはかなり違う。テレビ画面とコンピュータ画面を重ね合わせる，スーパーインポーズ機能。つまり，みゆきを見ながらインベーダゲームを同時に楽しめる。最近，このスーパーインポーズをつけたパソコンが他の会社から発売されたね。良いアイデアはどんどん採り入れて欲しいから別に構わないけどネ。それに，640×200ドットのカラー，タイリング，カラープライオリティー，パレット，アトリビュート，PCG，PSG機能そして，プリンタインターフェース，ジョイスティックインターフェースを内蔵というように，X1シリーズの血統は確実に受け継がれているんだ。そうそう，ボクのときはオプションだったグラフィックRAMは，本体に実装している。つまり，買ってすぐに640×200のカラーグラフィックが楽しめるようになった。また，ソフトウェア上の互換性だけど「フルコンパチブル」。カセットテープやフロッピーと媒体は異なるけど，プログラム自体は何の変更もなしに，そのままX1CであろうとX1Dでも使える。標準(付属)のBASICはCZ-8CB01(X1C)，CZ-8FB01(X1D)。

最後に，周辺機器を紹介するね。基本的にはX1シリーズ3機種に対し共通に使用できる。たとえ5½インチフロッピーでも。新しく追加されたものは，漢字プリンタやRS-232Cカードなど。どうですか。X1もシリーズ化したし，今後の活用例が大いに期待できるんじゃないだろうか。それから，X1シリーズ用にCP/Mが発売されるとのこと。ホビー，実務，CAI以外にれっきとしたソフト開発マシンになるのも近いうち。FORTRANやCOBOLだって走るんだ。おもしろいことになりそうだね。

（青木秀浩）

MZ-5500シリーズ



MZ-5500システム例。本体はドライブ2基のMZ-5521，カラーディスプレイは12インチ高解像度タイプMZ-1D11。これがチルトスタンドに乗っています。



新JIS配列のキーボード。ステップスカルプチャータイプの採用で，とても使いやすいのです。CRキーも大きくなり，ファクションは10個，シフトキー併用で20とおりの定義ができます。

MZシリーズに強力な16ビットパソコンが登場しました。数多くのメーカーから発表される数多くの新製品ラッシュにいささかうんざりされている読者の方も少なからずいらっしゃると思いますが，このMZ-5500はそんじょそこらのハンパな新製品をふき飛ばすだけのパワーを持つものだと思われます。

コンピュータの心臓部CPU8086の中で数の計算を行なうユニットのデータが16ビットである，つまり1度に2進数で16ケタの数の計算ができるものは，単に「16ビット」と省略して呼ばれますが，MZシリーズでは初めての16ビットパソコンです。

16ビットは8ビットに比べて単に一度に扱うデータ幅が2倍であるというだけならば，それほど大きな違いはないようにも思われますが，それだけにとどまらない能力差が存在します。直接アクセスできるメモリの量が大幅に増大することがまずあげられます。メモリRAMの値段がかなり安くなってきたため，これは大きな魅力と言えます。

さらにメモリとのデータ受け渡しの方法であるメモリアドレッシニクやインタラプト処理などでも大型コンピュータの構造を採り入れたアーキテクチャになっています。

このため16ビットパソコンと8ビットパソコンでは，システム全体を考えたときに，単なる演算能力を超えた差が歴然と存在するのです。

CPUを8ビットから16ビットにしただけならばこれだけの反響は呼びおこせなかったでしょう。次の2つの特徴はこのMZ-5500の最大のセールスポイントと言っても過言ではありません。

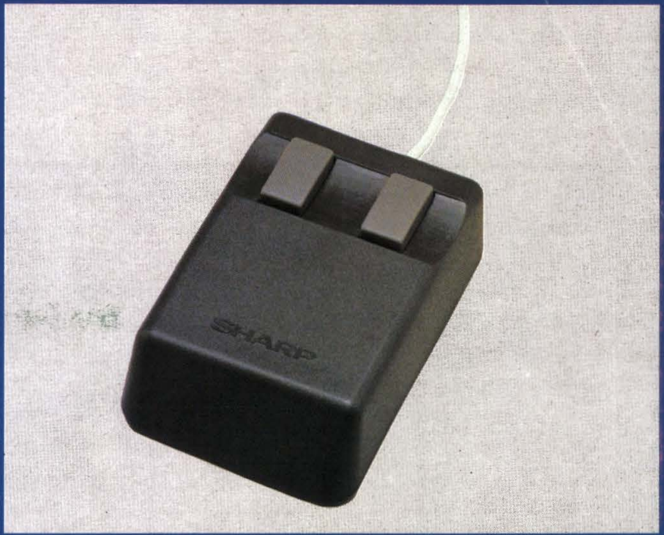
- ビットマップ方式，マルチウィンドウなどを含む強力なディスプレイ機能。
- 人間とコンピュータのインターフェースとして注目を集めているマウス。

あの有名な石田晴久教授は「日本にないのはマウスとビットマップディスプレイだけ」であり来年の夏ごろには「いっせいに発表されることになるのであろう」(『bit』8月号）と書かれていますが，1ヵ月もたたないうちに両者を装備（マウスはオプションだが）したパソコンが発表され，これには先生も驚かれたことでありましょう。

マルチウィンドウは好きな位置に好きな画面情報を表示できるもので，実際に種々の情報を記された書類の前で作業をしている感じを作り出すことが可能になります。またビットマップ方式の採用で画面のスクロールがスムーズになりました。



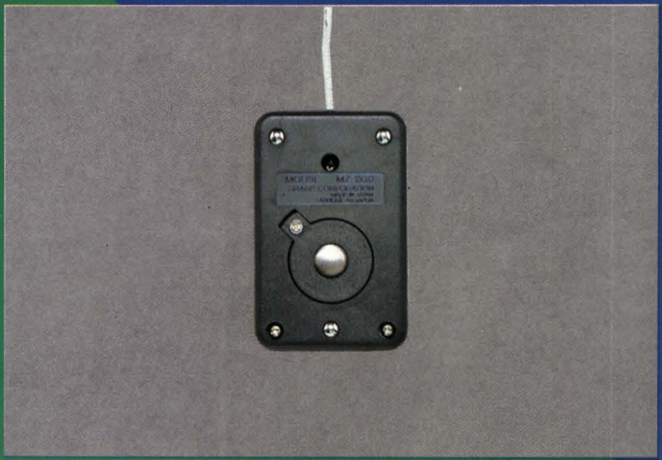
MZシリーズ用の最上位高品質インパクトドットマトリクス漢字プリンタ。16インチ用紙が使えるため132桁／行（漢字88桁／行）の印字ができます。漢字は24×24ドット構成ですから明朝体印字です。お値段54万円ナリ。



座標入力装置として，注目されているマウスです。国産ではこのMZ-5500用が初です。



本体に同梱のマニュアル。写真の他にもう一冊OWNER'S MANUAL，計4冊が付いてきます。サイズは，従来のMZ-2200用のより小さくハンディなＡ５判。これ全部内容を理解できれば，恐いものなしなんですけど…。



真ん中にトラックボールがあり，これを動かすことによりX-Y座標値を正確にとり込むことができます。

マウスはねずみのような形をした器具で画面上のカーソルを自由に移動させることができます。これによりまったく手元を見ずに画面だけ見ていればよいようなソフトも出現するでしょう。

従来のMZには本格的なビジネスへの応用にたえられるものはあまりなく，いわばホビー中心のイメージを持たれたのですが，5500の登場，しかもマウス付きの本格16ビットということでいよいよオフィスにもジワジワと進出するようになるでしょう。

マルチウィンドウやマウスという魅力的なハードがあってもソフトが遅れていればどうしようもないのですが，この5500では発売以前から各種アプリケーションが企画されており，日本語ワープロ，データベース，業務パッケージなど十分使えそうなものが数十種も発売予定されています。アメリカのように本体に数種のアプリケーションが装備されていればと思われますが，ソフト育成の姿勢が評価されますし，新たなハードを活用するためにも，市販のソフトにウェイトがかかってくることが予想されます。本誌では，登場するソフトを的確に評価し，ユーザーの判断材料となるようなデータを提供していきますので期待してよいでしょう。

目新しいハードですが，忘れてはならないのが従来のMZユーザーに対する配慮です。今までのMZシリーズ用のミニフロッピーMZ-80BFなどを所有している方のために，ミニフロッピーディスクドライブを外して，ドライブ２基内蔵のタイプから17万円価格と下げたバージョンも用意されています。17万円の差は大きく見えますが実装RAMの違いにもよっています。

またCP/M-86上のBASIC-1はMZ-2000系との互換性を考慮したものになっており，わずかの作業でMZ-2000/2200用のソフトを活用できるようになるでしょう。

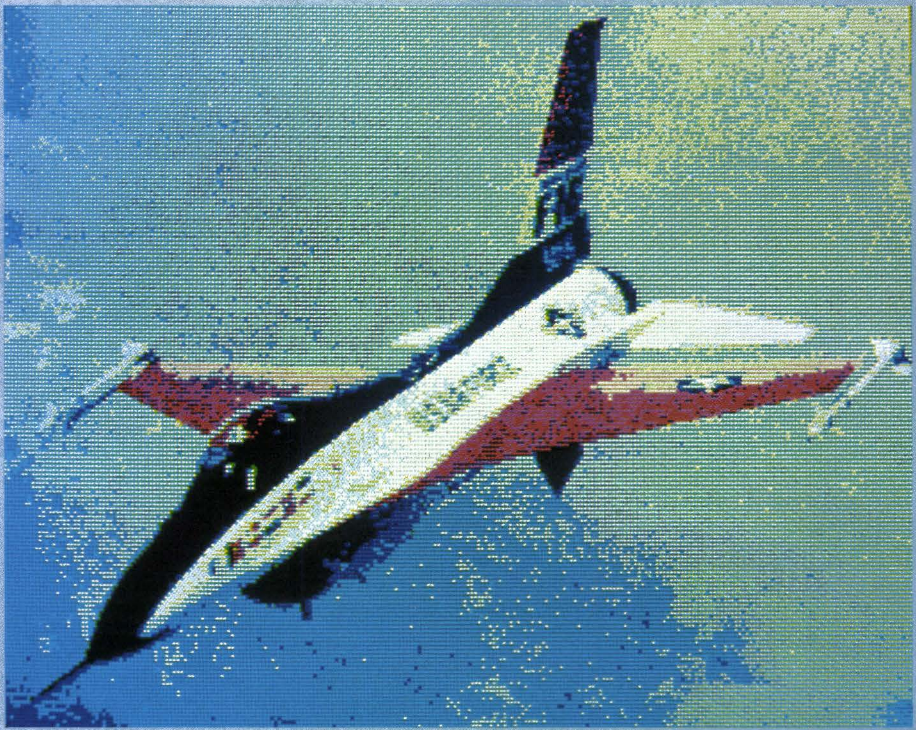
ビジネスだけでなくホビーも意識していることが８オクターブ３和音のPSGやカラーパレット機能などに表われています。アメリカのようにハイレベルのゲームがディスケットで次々に供給される時代を開くきっかけになると思われます。

単にCPUを16ビットにしたものではないことを読者の方も実際に自分の目で，自分の手で確かめられたらいかがでしょうか。　（有田隆也）

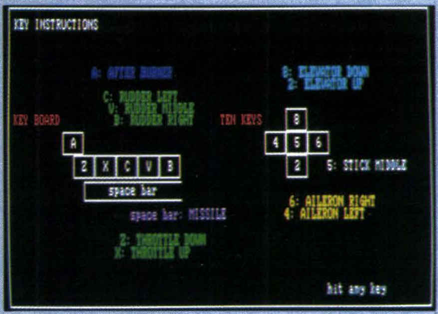
MZ-5500で「フライトシミュレータ」が走る!!

PC-9801のみに発売されていて，ＭＺシリーズにも……と悔し涙にくれていたキミ。ＭＺシリーズの16ビットパソコンの登場でこの大迫力ゲームが楽しめることになった。高速３Ｄグラフィックを駆使した本格的フライトシミュレータの興奮の一部を，今ここに紹介しよう。

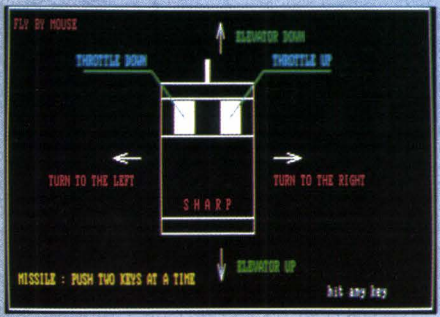
問い合わせ先：日本ソフトバンク出版部
定価6,800円(５インチFD版)



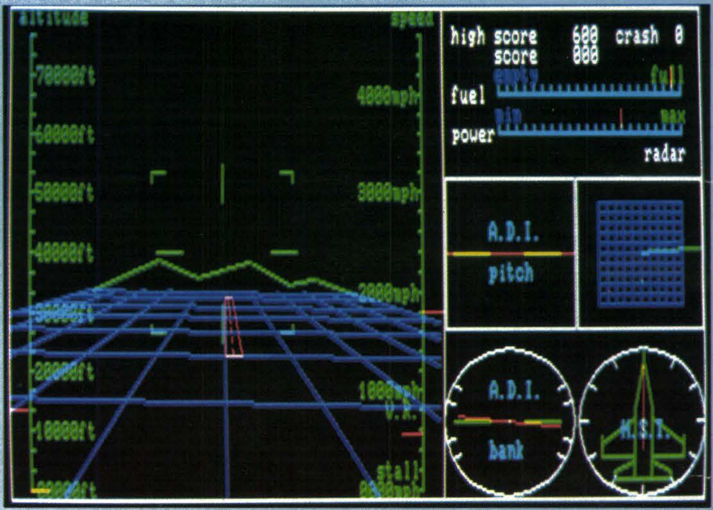
ビデオデジタイズした"F16"のデモ



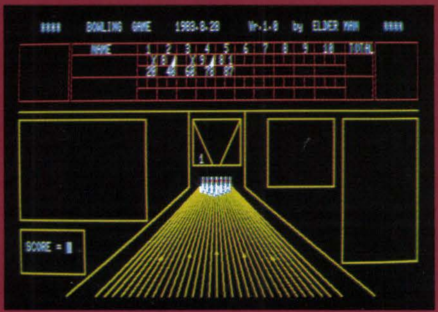
キーボード操作説明。12のキーを操るのだ



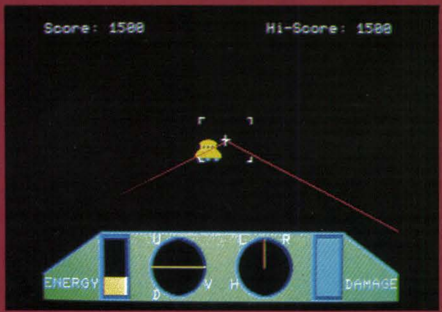
マウスを使うとこんなにかんたんになる



離陸！　スピードを上げて目標に向かえ



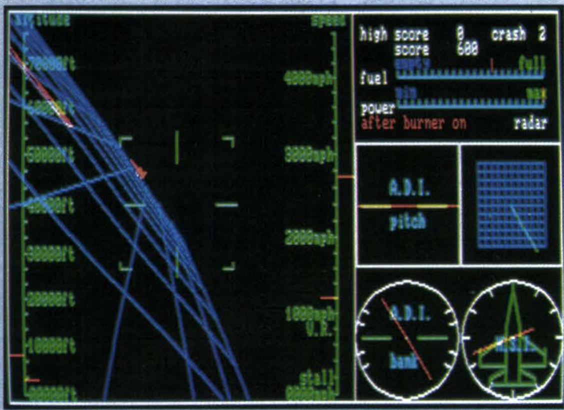
プロボーリング　本文89ページ



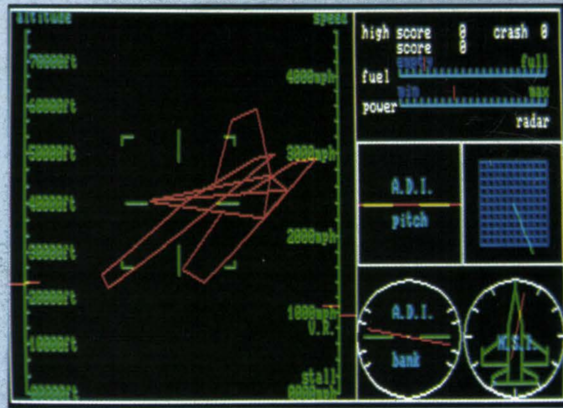
ＵＦＯ撃墜ゲーム　本文54ページ



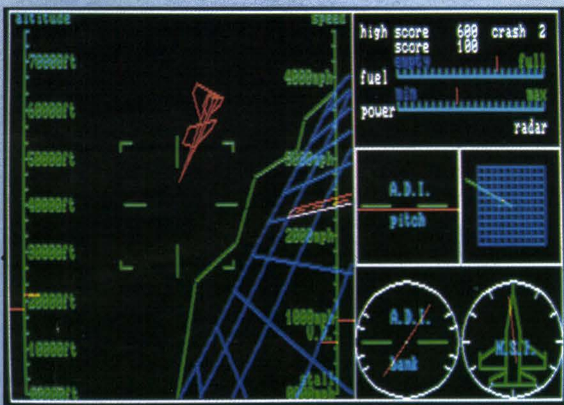
スカイダイバー　本文135ページ



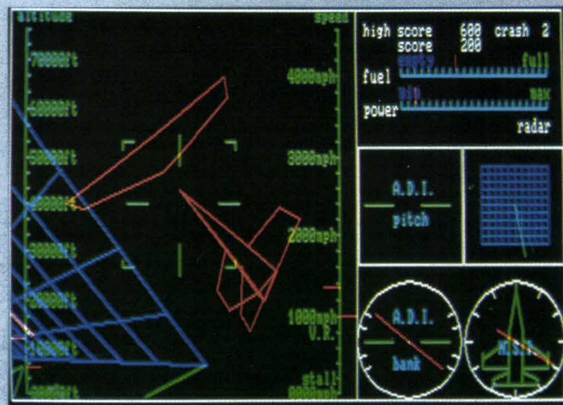
敵機発見。after burner on！



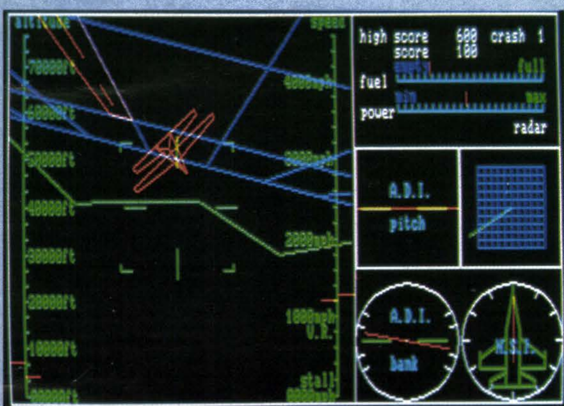
照準，スピードを合わせて，発射！



よし，下から追いあげるか



やった！　敵機撃墜



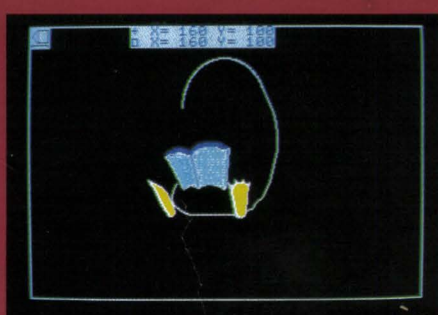
背面飛行で追う。もうじきだ



燃料補給のため着陸。excellent landing.



イタサンドリアス　本文135ページ





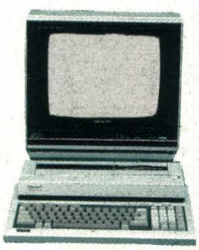
パソコンアニメーターで描いた“MZぺんぎん”です。本文134ページ

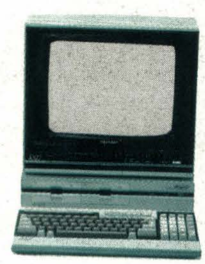
Oh!MZ 1983 NOV.

表紙原画／マジック・バス
表紙写真／浜崎　昭

特集　ユーティリティプログラム







充実の新鋭機

X1シリーズ/MZ-5500シリーズ

シャープの本格的な16ビットパソコンMZ-5500の登場はユーザー層のみならず業界各方面にもインパクトを与えたようです。今月号から何回かに分けてシリーズ紹介し、新鋭機MZ-5500の機能を探っていきたいと考えます。他方、昨年12月、テレビ・ビデオなどの映像と統合し、新しい時代を開く画期的な構想のパソコンテレビを発表し、着実にファン層を拡げてきたＸ１も、９月28日、普及タイプX1C、世界初の３インチコンパクトフロッピー標準実装のX1Dを発表して、シャープ旋風に輪をかけました。

パソコンテレビX1とシリーズ化
姉妹機X1C、X1Dが新登場

シャープは去る9月28日、独自のＶＩ（ビジュアルインテグレーション）構想に基づいた「パソコンテレビＸ１」をシリーズ化し、現行のＸ１（CZ-800C）に加え「Ｘ１Ｃ」「Ｘ１Ｄ」の2機種を発表しました。

Ｘ１シリーズは、従来の汎用パソコンと異なり、コンピュータがテレビ画面を併せて制御できるスーパーインポーズ機能をはじめ、高度なグラフィック、7番組予約などテレビコントロール機能を備えています。

現行のＸ１との互換性は完璧で、ソフトフルコンパチブル、すなわち今までＸ１用に発売されたすべてのソフトウエアが何の変更なしに利用できます。もちろん、ハードウエア上はメインメモリオールＲＡＭのクリーン設計。シリーズ用に同時発表された周辺機器も共用できますから、拡張性にすぐれたパソコンといえるでしょう。

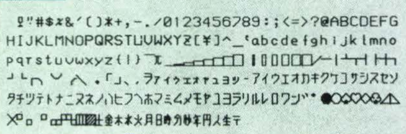
また、表示装置として欠かせないディスプレイテレビも2種同時発表されました。CZ-801S/Rは、シャドウマスクピッチをCZ-800Dの0.5㎜から更に0.45㎜ピッチとグレードアップしたファインピッチブラウン管を、そしてCZ-802S/Rはくし形フィルター回路内蔵のシャドウマスクピッチ0.40㎜にまで精細度をアップした新開発のディスプレイグレード中精細度ブラウン管を採用し、画像の高品位化を実現して価格も低く抑えています。もちろん、テレビコントロール、ビデオ入力、テレビ出力、ＲＧＢの各端子を備えていますから、従来のＸ１用CZ-800Ｄとともに自由に組み合せができるようになりました。

X1C カラープロッタプリンタも

Ｘ１Ｃは、キーボードとＣＰＵ部を一体型とし、ＭＺ-2200をさらにスリムにした個性的なデザインのコンパクト設計で、キーボードは手に馴じみやすいシリンドリカルスカルプチャで指の感触で入力が確認できるクリック付きを採用。Ｘ１独自のフルロジック高速データレコーダ(2,700ボー）を標準装備しています。また、オプションでカラープロッタプリンタを本体内に装着できます。このプロッタプリンタはＭＺ-700用と大変外観は似ており、好みの図形や文字を4色で表示しますが、Ｘ１Ｃ用はすべてのキャラジェネをプロッタに持たせることにより、グラフィックキャラクタ混在のリストが出力できます。

拡張Ｉ/Ｏは、本体がコンパクトなためＸ１のように内蔵できませんが、専用の拡張I/Ｏボックスがオプションで用意されていますので、フロッピーディスクインターフェースボードや漢字ＲＯＭボードが収納でき、拡張性にもすぐれています。

図１　プロッタプリンタの印字例



X1D コンパクトフロッピーを内蔵

Ｘ１Ｄは現行のＸ１と同様のオーディオラックマウントサイズで統一したＣＰＵ、キーボードのセパレートタイプです。そして驚いたことに新開発の両面倍密度設計の3インチコンパクトフロッピーを１基本体内に標準装

表１　パーソナルコンピュータ(CZ-801C/802C)仕様

項目		仕様
CPU		Z-80A　4MHz 80C48　キースキャン用 80C49　テレビコントロール、カセットコントロール用 8877A　フロッピーコントロール用
ROM		ＩＰＬ用ＲＯＭ　4Ｋバイト キャラクタジェネレータ用ＲＯＭ　2Ｋバイト
RAM		プログラム用ＲＡＭ　64Ｋバイト テキスト用ＲＡＭ　4Ｋバイト ユーザー定義キャラクタジェネレータ用RAM　6Ｋバイト グラフィック用ＲＡＭ　48Ｋバイト
表示能力	テキスト表示	80字×25行，40字×25行（選択可能） 反転文字，点滅文字，縦・横← 縦横２倍文字可能 カラー８色（文字毎に色指定可能）
	カラーグラフィック表示 ※右のいずれかの画面を選択可能	640×200ドット　1画面 320×200ドット　2画面 カラー８色（ドット単位に色指定可能）
		640×200ドット　3画面 320×200ドット　6画面 カラー８色（1画面単位に色指定可能）
	画面合成 ※右のいずれかの画面を選択可能	テキスト画面とグラフィック画面 テキスト画面とグラフィック画面とテレビ画面 テキスト画面とグラフィック画面とビデオ画面
	プライオリティ機能	テキスト・グラフィック画面の優先順位がつけられる
	バックグランドカラー	８色指定可能
	ビデオ出力	R・G・Bセパレート出力方式

項目	仕様
コンパクトフロッピーディスクドライブ	両面倍密度　80トラック フォーマット時　320Ｋバイト ３インチコンパクトフロッピーディスク使用
カセットレコーダ	フルロジック電磁メカ式内蔵（X1C） データ転送方式―シャープＰＷＭ方式 転送速度―2700ボー オーディオ用カセットテープ使用可能
サウンド出力	８オクターブ　３和音
音声出力	８㎝丸型スピーカー　300mW
プリンタインターフェース	セントロニクス社仕様に準拠　８ビットパラレル
カラープロッタプリンタ(X1C)	TTLレベル，８ビットパラレル
ジョイスティックインターフェース	アタリ社仕様・２個使用可能
拡張I/Oポート(X1D)	本体内に４ポート収納可能（オプション）
I/O BUS 端子 (X1C)	拡張I/Oボックス、CZ-81EB接続用端子
時計機能	内蔵（X1Dは、内蔵ニッカド電池にてバックアップ）
電源	AC　100V±10%　50／60Hz
消費電力	49W(X1D)　30W(X1C)
使用条件	使用温度　10～35℃ 使用湿度　35～75%
外形寸法	本体　幅39.0㎝，奥行33.2㎝，高さ10.8㎝ キーボード　幅39.1㎝，奥行18.9㎝，高さ5.3㎝(X1D) 本体　幅42.0㎝，奥行34.4㎝，高さ11.4㎝(X1C)
重量	本体 6.2㎏，キーボード 1.35㎏（X1D） 本体 5.0㎏（X1C）

備しています。もちろん世界初。記録容量は5 1/2インチミニフロッピー（両面倍密度）と同じ320Kバイトを実現していますから、手のひらサイズで各種ファイルを手軽にしかも高速で入出力することができます（本体内にはあと1基増設が可能）。さらに、本体内にインターフェースを内蔵していますので、従来の5 1/2インチミニフロッピー（CZ-800F）やカセットデータレコーダが接続できます。

拡張I/Oポートは、X1と同様CPU本体に収納する省スペース設計です。

CP/M ワードマスターをバンドル

X1シリーズにもうひとつのお知らせ。このX1C/Dの発売と同時に、シャープからX1シリーズ用にアメリカ、デジタルリサーチ社製のCP/M-80 Ver2.2が発売されます。3インチコンパクトフロッピーがX1Dに装備されたことでもあり、X1シリーズ用CP/Mは5 1/4 インチ版（CZ-5CPM）に加え、コンパクトフロッピー（CZ-3CPM）でも供給されます。3インチ版CP/Mは業界初の最小のメディアで発売されることになります。ここで改めてCP/Mの必要性を述べることはないと思いますが、CP/Mが走ると、FORTRAN,COBOL,Ada,etc,の高級言語が利用できるようになります。そして、X1-CP/MはX1シリーズの特徴を生かすため、PSG,グラフィック、大容量RAM、RS-232Cなどを特にサポートしていますから、かなり強力なCP/Mマシンとなることでしょう。

また、強力なスクリーンエディタとして定評のあるマイクロプロ社のWordmaster（ワードマスター）をバンドル（付属）しています。

ソフトウェア

X1シリーズのラインナップが完成したことで、ソフトウェアの大幅な増が早くも期待できそうです。現在500余本という流通ソフトウェアに加え、ホビー・実務を始め多様化するニーズに応えるべくいろいろな分野のソフトウェアが発売されます。特に3インチコンパクトフロッピーの採用により、多重のデータを一度にしかも高速で扱えるため、本格的な日本語処理（日本語ワープロ）やビジネスソフトが発売を控えています。シャープも今回はソフトウェア開発にかなり力を注いでおり、年内にも100余本新たにX1シリーズのライブラリーとして登場するそうです。

▶価格表

品　名	パーソナル コンピュータ	ディスプレイ テレビ	パーソナル コンピュータ	ディスプレイ テレビ
機種名	CZ-801C S/R	CZ-801D S/R	CZ-802C S/R	CZ-802D S/R
標準価格	119,800円	99,800円	198,000円	128,000円
愛称	パソコンテレビ X1C		パソコンテレビ X1D	
発売日	10月11日			

図2　X1C, X1Dシステム構成図　黒色文字はX1C、青色文字はX1D、他は共通

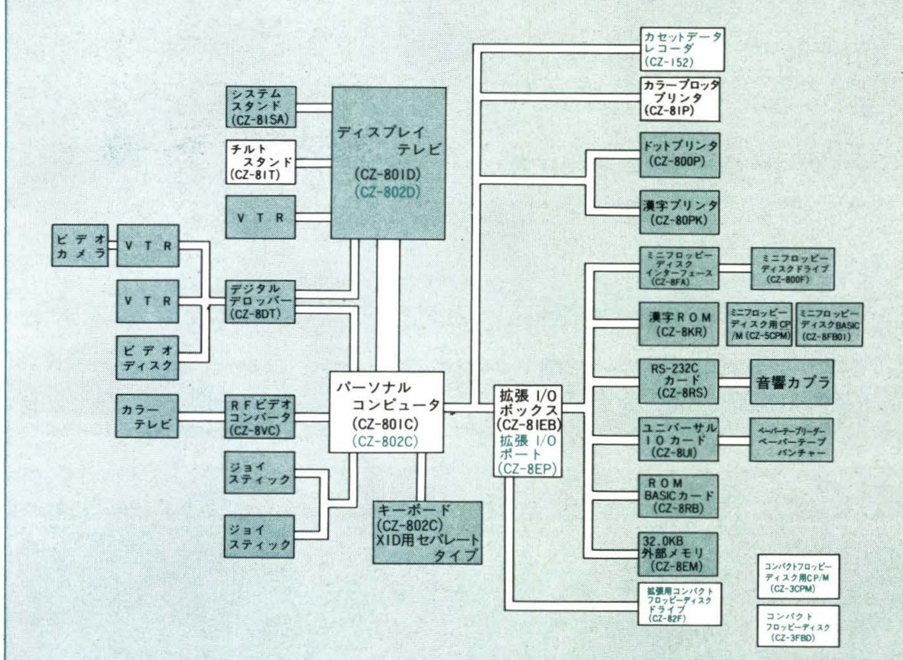


表2　周辺機器の状況および共通性

（★：新製品）

	品名	機種名	標準価格	共用可能機種 X1C (CZ-801C/D)	X1D (CZ-802C/D)	X1 (CZ-800C/D)
プリンタ	ドットプリンタ（シルバー／レッド）	CZ-800P(S/R)	¥142,800	○	○	○
	★漢字プリンタ（シルバー／レッド）	CZ-80PK(S/R)	¥123,800	○	○	○
	★CZ-801C内蔵用カラープロッタプリンタ（シルバー／レッド）	CZ-81P(S/R)	10月下旬発売予定	○	—	—
ディスクユニット	ミニフロッピーディスクドライブ	CZ-800F	¥198,000	○	○	○
	ミニフロッピーディスクインターフェース	CZ-8FA	¥ 24,000	○	内蔵	○
	★ミニフロッピーディスクドライブ（シルバー／レッド）	CZ-801F(S/R)	¥198,000	○	○	○
	★CZ-802C内蔵用コンパクトフロッピーディスクドライブ（シルバー／レッド）	CZ-82F(S/R)	¥ 59,800	—	○	—
ディスク	ミニフロッピーDISK BASIC	CZ-8FB01	¥ 6,800	○	○	○
	★ミニフロッピーディスク用CP／M	CZ-5CPM	同時発売	○	○	○
	★CZ-802C用コンパクトフロッピーディスク用 CP／M	CZ-3CPM	同時発売	—	○	—
	★CZ-802C用3インチコンパクトフロッピーディスク（ブランク）	CZ-3FBD	¥ 1,300	—	○	—
拡張装置	拡張 I／O ポート	CZ-8EP	¥ 11,800	—	○	○
	★CZ-801C用 拡張 I／O ボックス（シルバー／レッド）	CZ-81EB(S/R)	同時発売	○	—	—
基板	グラフィック RAM	CZ-8GR	¥ 32,000	内蔵	内蔵	○
	漢字 ROM	CZ-8KR	¥ 38,000	○	○	○
	ROM BASIC カード	CZ-8RB	¥ 19,800	○	○	○
	320KB 外部メモリ	CZ-8EM	¥ 88,000	○	○	○
	ユニバーサル I／O カード	CZ-8UI	¥ 14,800	○	○	○
	★RS-232C カード	CZ-8RS	¥ 29,800	○	○	○
テレビ台	★チルトスタンド（シルバー／レッド）	CZ-81T(S/R)	¥ 8,500	○	○	—
	★システムスタンド	CZ-81SA	¥ 5,300	○	○	○
ラック	システムラック1（シルバー／木目）	CZ-8SR1(S/W)	¥ 29,800	—	○	○
	システムラック2（シルバー／木目）	CZ-8SR2(S/W)	¥ 29,800	—	○	○
その他ハードウェア	デジタルテロッパー	CZ-8DT	¥ 89,800	○	○	○
	RFビデオコンバータ	CZ-8VC	¥ 15,800	○	○	○
	カセットデータレコーダ	CE-152	¥ 19,800	内蔵	○	内蔵

華麗なシステムで迫る 新鋭機MZ-5500シリーズレポート 1

前回もご報告したようにMZ−5500シリーズは、実装メモリ容量、ミニフロッピーの基数により3タイプ用意されています。もっとも安いタイプMZ-5501は、フロッピーディスクドライブが付いていませんが、これはカセットバージョンということではなく、姉妹機MZ-2000用のミニフロッピーディスクドライブ(MZ-1F02,-1F07,-80BF)を持っている人用のモデルということです。なお、MZ-5500用にもデータレコーダが用意されていますが、外部記憶装置として使えるのではなく、MZ-2000などのアプリケーションテープの読み出し専用です。

それでは、ハードウェアから紹介しましょう。

ハードウェア

システムの心臓部であるCPUには、インテル社の16ビットマイクロプロセッサ8086(動作クロックは約5MHz)を使用しています。従来の8ビットCPUに比べ、大容量のメモリが一度に扱えるほか、高速処理ができるのです。さらに、キーボード部には、専用のCPU(80C49)を搭載することによりメインCPUの負担を軽減しています。また、オプションである数値演算プロセッサ(i8087)を追加すれば、四則演算はもとより各種数値演算が高精度で高速処理(今までプログラム上で行なっていた処理がハードウェア上でできると考えるとよい)が可能となります。

グラフィックは、96KバイトのV-RAMで最大640×400ドット1画面、ドット単位で色指定が可能です。この制御には、専用LSI(GDC、μPD7220)を使用しているため、マルチウインド機能などグラフィック処理に威力を発揮します。

メモリ容量は、16ビットCPUになったおかげで、MZ-5521で256Kバイト(MZ-5511およびMZ-5501は128Kバイト)を実装。本体メインボードで256Kそして拡張スロットに256K、合わせてなんと512Kバイトまで拡張できます。

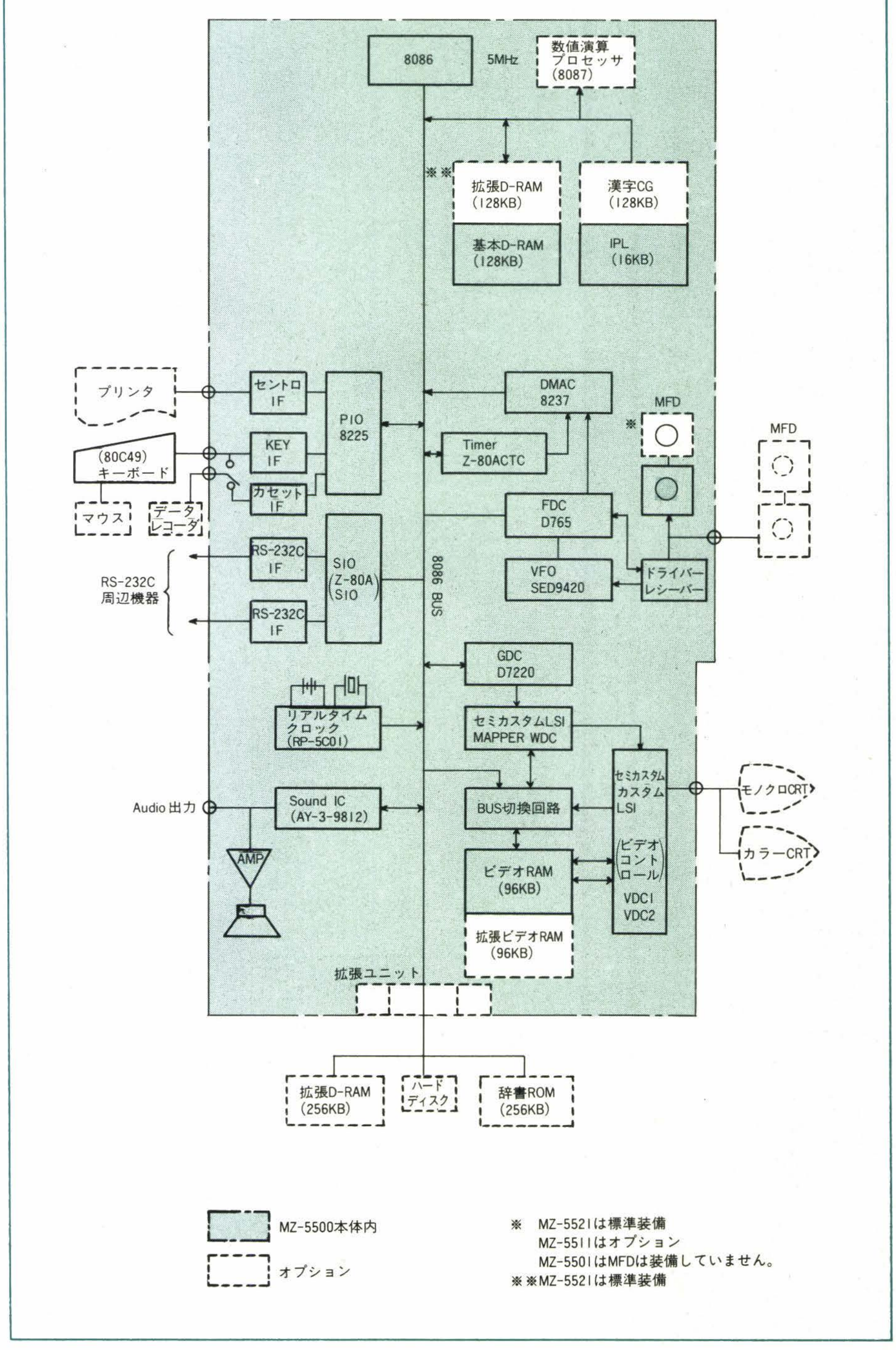
その他特徴として、PSG(プログラマブル・サウンド・ジェネレータ)を内蔵しています。MZシリーズは、以前から音楽機能がサポートされていましたが、単音しか出ず、今となってはさしてめずらしいものではなくなってしまいました。この新鋭機MZ-5500ではこのPSGを搭載することにより、3重和音8オクターブまでの音楽演奏ができます。

周辺のインターフェース類は、プリンタ(セントロニクス社仕様準拠)、シリアルポート(RS-232C、2ポート)、マウス、CMTなどすべて実装しています。

ソフトウェア

MZ-5500シリーズは、今までの中途半端なカセットベースシステムとは異なり、完全なディスクベースとなっています。したがって、基本的な入出力を管理するOSは欠かせません。シャープは、16ビットのDOSとしてスタンダード的な存在となりつつあるCP/M-86を採用しました。さらにオプションですが、このほかに漢字CP/M-86、MS-DOSも発売予定となっていますから、これらの

図1　システム構成図



DOSの管理下で動作する市販のアプリケーションが利用できますのでDOSレベルでのソフトウェア環境は申し分ありません。

さて、皆さんの一番の関心ごとといえばやはりBASICは何かということではないでしょうか。シャープでは、サポートするDOSに合わせ、3種のBASICの発売を決めています。そのうち、BASIC-1(CP/M-86用)およびBASIC-2(漢字CP/M-86用)はいわゆるシャープBASIC。GW-BASIC(MS-DOS用)はマイクロソフト系のBASICとなっています。BASIC-1とBASIC-2の違いは、漢字入出力の強化および数値演算プロセッサ(8087)サポートなどくらいで、どうやら基本的には変わらないようです。また、GW-BASICは、年内にも発表されると思われる汎用低価格パソコン用BASIC、MSX-BASICとの上位コンパチビリティを備えたマイクロソフト社最新のBASICで、主流といわれる8ビット用MBASICの上位バージョンと考えて支障ないでしょう。

標準装備のBASIC-1(2Z-012)は、MZ-2000用のBASIC MZ-2Z001をMZ-5500に移植し、拡張した感じで、慣れ親しんだシャープBASICがこのMZ-5500でも気軽に楽しめそうです。新しく追加された命令が15と、極力押さえられているのもBASICレベルの互換性を尊重した結果といえるでしょう。

このように、数々の魅力を秘めたMZ-5500。今号が発売されるころ開催されるデータショウ'83でも多くの人の熱い視線を集めることでしょう。

本誌でも今後、あらゆる角度から記事を掲載して、MZ-5500の活用法、活用例を紹介していく方針とのことで、楽しみです。

表1　型式と価格

	ミニフロッピーディスクドライブ	システムRAM	価　格	発　売
MZ-5521	2基内蔵	256Kバイト	388,000	発売中
MZ-5511	1基内蔵	128Kバイト	288,000	発売中
MZ-5501	なし	128Kバイト	218,000	10月中旬

＊MZ-5501は、外部にミニフロッピーディスクドライブ(MZ-1F02, 1F07, 80BF)が必要です。

表4　仕様

形　名…………MZ-5501/MZ-5511/MZ-5521
CPU……………8086(16ビット：クロック5MHz)
　オプション：8087(16ビット演算専用チップ)
ROM……………IPL他(16KB)
　オプション：漢字CG(128KB)、辞書ROM(256KB)
RAM……………システムRAM(128/256KB)
　システム・プログラム・データ用
　ビデオRAM(96KB)
　オプション：拡張システムRAM(128/256KB)、合計最大512KB、拡張ビデオRAM(96KB)、合計最大192KB
インターフェース…プリンタ インターフェース(セントロニクス方式準拠)
　CRT インターフェース(コンポジット/RGB)
　ミニFD インターフェース(最大4基コントロール可)
　RS-232C インターフェース(2ポート)
　カセット インターフェース(MZ-2000系BASICプログラムコンバート用)
ミニFD………両面倍密度薄型FD
　容量：320KB(1ドライブ)
　トラック数：40トラック
　セクター数：16セクタ
　記録媒体：5¼インチ・ディスク
　本体内2基まで内蔵可、0基/1基付/2基付の3タイプ
ソフト…………標準OS：CP/M-86
　BASIC：BASIC-1　以上、本体に付属
その他…………サウンド機能(3和音・8オクターブ)
　カレンダー・クロック機能(リアルタイムクロック、バッテリーバックアップ)
使用素子………LSI、その他
動作電源………AC100V、50/60Hz
消費電力………110W
使用温度………10℃～35℃
使用湿度………20%～80%(ただし、結露しないこと)
外形寸法………幅430mm×奥行410mm×高さ123mm
重　量…………13.7kg(MZ-5501)
　15.2kg(MZ-5511)
　16.7kg(MZ-5521)
付属品…………MANUAL(4冊)、マスターディスク、ファンクションラベル、ドライブ番号シール、電源コード、保証書、ほか
(キーボード)
配　列…………JIS配列
使用素子………LSI(80C49)、その他
使用温度………10℃～35℃
使用湿度………20%～80%(ただし、結露しないこと)
外形寸法………幅465mm×奥行190mm×高さ40mm
重　量…………1.7kg

図2　ソフトウェア

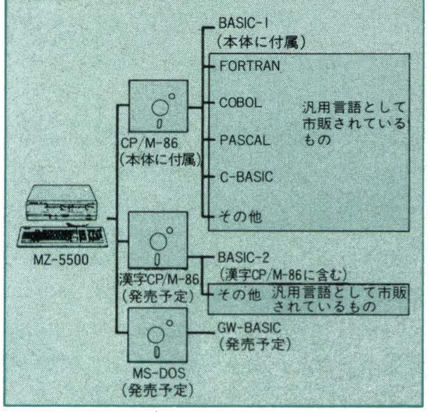
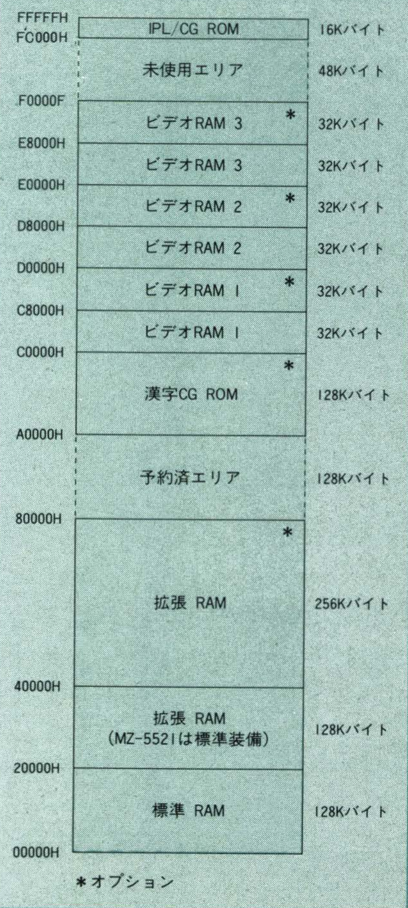


図3　MZ-5500シリーズのメモリマップ



MZ-2200/2000
BASICユーティリティ
――リナンバ，アペンド，デリート機能――

小笠原吉義

はじめに

MZ-2000でBASICのプログラムを作るときに有用な，リナンバ，アペンド，デリートなどの機能を持つユーティリティを作りましたので紹介したいと思います。

BASICインタプリタ自体の内部ルーチンを使用しているので，MZ-1Z001 V1.0Cにだけ有効で，MZ-80 BやディスクBASICでは動きません。ただし，使用した内部ルーチンについてはあとで説明しますので，それぞれのBASICのサブルーチンの入口がわかれば，比較的容易に移植できるでしょう。

機能

①リナンバ（Renumber）

BASICプログラムの行番号を，指定した範囲について付け換えます。もちろん，同時にGOTO文などの飛び先も変更します。

②デリート（Delete）

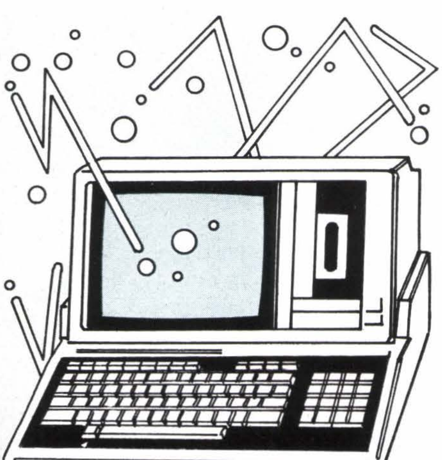
行番号で指定した範囲内のBASICプログラムを削除します。

③アペンド（Append）

すでにロードされているBASICプログラムのあとに，カセットテープから新たに他のBASICプログラムを読み込んで付け加えます。ただし，行番号の並べ換えの機能は含まれていません。

④ソート（Sort）

リナンバやアペンドによって行番号のバラバラになったBASICプログラムを，行番号が昇順(小→大)になるように並べ換えます。



プログラムの入力

リスト1の16進数をモニタのMコマンドなどで打ち込んでください。先頭番地は必ずEA00番地にしてください。入力し終わったら，Sコマンドでセーブをすることをお忘れなく。

参考にアセンブルリストを示しておきます（リスト2）。

プログラムの使い方

LIMIT＄E9FF [CR] でBASICのテキスト領域を制限してから，処理したいBASICプログラムをロードしておきます。そのあと，このプログラムをモニタのLコマンドで読んでおいてからモニタのJコマンドで＄EA00番地へジャンプしてください。

メニューが表示されます。メニューの番号を入力してください。

①リスタート

メッセージに従って，リナンバの範囲の開始行番号，終了行番号，新しい開始行番号，増分を順に入力します。メッセージの中の（Def：…）はデフォルト値で，単に [CR] キーを押したときに入力される値です。全部 [CR] キーだけを入力すると，BASICプログラム全体の行番号を10，20，30……と10ごとに付け換えます。

②デリート

メッセージに従って，削除したい部分の先頭行番号と終了行番号を入力してください。両方を省略するとNEWと同じになってしまうので，エラーとみなしています。

③アペンド

付け加えたいBASICプログラムの入ったテープをカセットにセットして，ファイル

リナンバの使用例

```
** MONITOR MZ-1Z001M **

  BASIC interpreter  MZ-1Z001 V1.0C

  Copyright 1982 by SHARP Corp.

 43500 Bytes
Ready
LIMIT $CFFF
Ready
LOAD
FOUND "ﾁﾃｲ ｻｲﾀﾞｲ ﾉ ｻｸｾﾝ"
LOADING "ﾁﾃｲ ｻｲﾀﾞｲ ﾉ ｻｸｾﾝ"
Ready

MON
*L
FILE NAME:
LOADING RENUM.OBJ
*J
J-ADR.$EA00
***(MENU)***
1.<RENUMBER>
2.<DELETE>
3.<APPEND>
4.<SORT>
5.END
#ﾅﾝﾊﾞﾝ?
1
<RENUMBER>
*START LINE#(Def: 0)?
1000
*END LINE# (Def:max)?

*NEW NUMBER (Def:10)?
2000
*STEP NUMBER(Def:10)?
50
 129 LINES.--<END>
***(MENU)***
1.<RENUMBER>
2.<DELETE>
3.<APPEND>
4.<SORT>
5.END
#ﾅﾝﾊﾞﾝ?
5
Ready
```

アペンドの使用例

```
#ﾅﾝﾊﾞﾝ?
3
<APPEND>
FILE NAME:
LOADING HEIANKYO
<END>
```

名を入力してください。ファイル名は省略しても構いません。元のプログラムと行番号が重複する場合は，あらかじめリナンバで重複しないようにしておいてください。

④ソート

メニューでソートを選ぶと，直ちにソートを開始します。リナンバと組み合わせて使うことで，一連のプログラム列を移動させるといった使い方ができます。

⑤エンド

BASICインタプリタ($1300番地)へ制御を戻します。

コメント

2～3回使ってみれば使い方がわかると思います。いろいろ工夫して活用してみてください。一応デバッグは済み，今のところ完動しています。

EE00番地からあとの領域は，ワーク・エリアとして使用しています。

一般にBASICインタプリタからマシン語を使用するときには，スタックエリアなどの関係で LIMIT 文が必要です。お忘れのないように。

使用したBASICインタプリタのルーチン

①CALL 1A92H

(HL)と(HL＋1)にAレジスタの内容をストア。HL=HL＋2になる。

②CALL 1A91H

(HL)と(HL＋1)を2バイト0クリア。HL=HL＋2になる。

③CALL 1988H

HLで示される行番号を先頭からサーチ。

1)あるとき：ZF＝1，「HL＝その行番号のアドレス」にして(アドレス＋2)へリターン。

2)行ポインタが0で見つからないとき：リターン・アドレスの内容の番地へリターン。

3)行番号＞サーチ番号となって見つからないとき：ZF＝0にして（リターン・アドレス＋2）へリターン。

(使用例)

CALL 1988H

DEFW a→ポインタ=0のときの飛び先

NEXT命令（Zフラグ判定）

④CALL 193FH

HLの示す行の次の行ポインタをHLレジスタにセット。行番号をDEレジスタにセット。行ポインタが0のとき，ZF＝1。

⑤CALL 1845H

HLとDEの大小比較。

HL＝DEのとき:ZF＝1，その他ZF＝0

HL＜DEのとき:CF＝1，その他CF＝0

⑥CALL 13A8H

HLの示す行を1行デリート。

⑦CALL 19CAH

行の長さを計算する。入力：HL＝行先頭アドレス。出力：BC＝バイト数(0Dコードを含む)。

⑧CALL 197AH

行インサートのためのテキストの移動。

⑨CALL 13B2H

ポインタとテーブル修正。

⑩CALL 1BEFH

BASICテキストの各ラインのポインタ部の内容を，ライン長に変換する。

⑪CALL 1960H

インサートのためのテキスト・ムーブ。

入力:DE＝転送元アドレス，BC＝転送バイト数。

⑫CALL 1B96H

CALL 1BEFHの逆変換。

⑬CALL 18A7H

10進数の文字列を2進数にする。

入力：文字列のアドレスをHLレジスタ。

出力：2進数をDEにセット。

正常なとき：リターン・アドレス＋2へリターン。

エラーのとき：リターン・アドレスの内容の番地へリターン。

(使用例)

LD HL，文字列アドレス

CALL 18A7H

DEFW a→エラーのときの飛び先

NEXT命令

⑭CALL 18D4H

HLレジスタの2進数をDEレジスタの示すアドレスから10進ASCIIコードに変換してセット。

入力：HL＝2進数，DE＝格納先頭アドレス。

出力：(DE) DEはここをポイントとする。

20	…10進コード……	0D

C＝桁数

⑮CALL 1867H

文字のチェック

(使用例)

CALL 1867H

DEFB a

DFFW x

1：NEXT命令

(HL)ブランク・スキップを行ない，次の内容がaであるかどうかを調べる。YESのとき，(HL)ブランク・スキップ後，1へリターン。NOのときxへリターン。

⑯データ・エリア

$545C:BASICテキスト先頭アドレス

$4D0D:BASIC用入力エリア

$4E0E:行編集用作業エリア

リスト1

```
** Oh!MZ Check sum system V1.0 **

ADR. +0 +1 +2 +3 +4 +5 +6 +7 :sum.
EA00 18 08 00 00 FF FF 0A 00 :2.28
EA08 0A 00 ED 7B 0A 4D CD 4F :2.E5
EA10 ED CD 54 ED CD 59 ED CD :5.DB
EA18 5E ED CD 63 ED CD 68 ED :5.8A
EA20 CD 6D ED 11 0D 4D CD C5 :4.24
EA28 05 3A 0D 4D FE 31 CA 4A :2.DC
EA30 EB FE 32 CA 47 EA FE 33 :5.47
EA38 CA E9 EA FE 34 CA 90 EA :6.13
------------------------------------22CC
EA40 FE 35 CA 00 13 18 D9 11 :3.12
EA48 93 ED CD 70 ED CD D0 EC :6.33
EA50 38 F5 7E 2B B6 20 0C 3E :2.F6
EA58 FF CD 92 1A 2A 02 EA 7C :4.0A
EA60 B5 28 E4 21 00 EE CD 91 :4.2E
EA68 1A 2A 02 EA CD 88 19 8A :3.28
EA70 EA E5 CD 3F 19 2A 04 EA :4.0C
EA78 CD 45 18 E1 38 0C CD A8 :3.C4
------------------------------------1F6B
EA80 13 2A 00 EE 23 22 00 EE :2.5E
EA88 18 DF CD 24 ED C3 00 EA :4.82
EA90 11 A9 ED CD 70 ED 21 00 :3.F2
EA98 EE CD 91 1A 21 5C 54 CD :4.04
EAA0 3F 19 28 3F E5 D5 CD 3F :3.85
EAA8 19 28 36 E1 EB CD 45 18 :3.6D
```

```
EAB0 E1 30 EC CD CA 19 E5 11 :4.A3
EAB8 0E 4E ED B0 E1 CD A8 13 :4.62
------------------------------------1ECD
EAC0 2A 10 4E CD 88 19 00 EA :2.E0
EAC8 22 0E 4E EB 21 0E 4E CD :2.B3
EAD0 CA 19 CD 7A 19 CD B2 13 :3.D5
EAD8 2A 00 EE 23 22 00 EE 18 :2.63
EAE0 BB F1 F1 CD 24 ED C3 00 :5.3E
EAE8 EA 11 9E ED CD 70 ED CD :5.7D
EAF0 EF 1B 3E D8 32 D9 01 32 :3.5E
EAF8 DA 01 3E AF 32 ED 01 3E :3.26
------------------------------------1D0A
EB00 C9 32 EE 01 CD CB 01 3E :3.C1
EB08 38 32 D9 01 3E E2 32 DA :3.70
EB10 01 3E CD 32 ED 01 3E B2 :3.1C
EB18 32 EE 01 38 1A 2A 52 11 :2.00
EB20 2B 2B 4D 44 ED 5B 4E 52 :2.CF
EB28 1B 1B CD 60 19 CD 9A 1A :2.FD
EB30 ED 53 54 11 CD B2 02 CD :3.F3
EB38 96 1B 06 05 CD 14 0F 10 :1.BC
------------------------------------17C8
EB40 FB 11 49 ED CD 70 ED C3 :5.2F
EB48 00 EA 11 86 ED CD 70 ED :4.98
EB50 CD D0 EC 38 F5 7E 2B B6 :5.15
EB58 20 05 3E FF CD 92 1A 11 :2.EC
EB60 BE ED CD 70 ED 21 06 EA :4.E6
EB68 CD E6 EC 38 F2 7E 2B B6 :5.28
EB70 20 02 36 0A 11 D4 ED CD :3.01
EB78 70 ED 21 08 EA CD E6 EC :5.0F
------------------------------------23E6
EB80 38 F2 7E 2B B6 20 02 36 :2.E1
EB88 0A ED 4B 06 EA 21 02 EE :3.43
EB90 E5 2A 02 EA CD 88 19 00 :3.69
EB98 13 CD 3F 19 28 0F CD C7 :3.03
EBA0 EC 38 0A E3 73 23 72 CD :3.E6
EBA8 BA EC E3 18 EC E1 3E FF :5.AB
EBB0 CD 92 1A 21 5C 54 E5 18 :3.47
EBB8 05 E1 CD 3F 19 E5 E5 CD :4.A2
------------------------------------1E0A
EBC0 84 EC CD 84 EC 7E FE 0D :5.36
EBC8 CA 61 EC FE 22 28 30 FE :4.8D
EBD0 80 20 EF CD 84 EC FE 86 :5.50
EBD8 28 32 FE 8C 28 2E FE 8D :3.C5
EBE0 28 2A FE 8E 28 26 FE 9C :3.C6
EBE8 28 22 FE 80 28 04 FE 81 :3.73
EBF0 28 2A FE 8E 28 26 FE 9C :3.C6
EBF8 28 22 FE 80 28 04 FE 81 :3.73
------------------------------------214A
EC00 20 D0 CD 84 EC FE 0D 28 :4.60
EC08 58 FE 3A 28 C5 18 F3 CD :4.55
EC10 84 EC FE 0D 28 4B FE 22 :4.0E
EC18 20 F5 18 B6 CD 84 EC D5 :4.F5
EC20 CD A7 18 44 EC 7A B3 28 :4.11
EC28 1B 22 00 EE D5 21 02 EE :3.11
EC30 7E 4F 23 A6 3C 28 16 79 :2.89
EC38 BB 20 04 7E BA 28 08 23 :2.6A
------------------------------------1DCD
EC40 23 23 18 EC D1 18 8E 23 :2.E4
EC48 5E 23 56 EB E3 E1 D1 CD :5.24
EC50 D4 18 2A 00 EE CD 67 18 :3.50
EC58 2C D5 EB 3E 2C 12 13 18 :2.93
```

```
EC60 BE CD 84 EC CD ED 14 CD :5.96
EC68 EB 14 E1 E5 CD A8 13 E1 :5.2B
EC70 E5 22 0E 4E EB 21 0E 4E :2.CB
EC78 CD CA 19 CD 7A 19 CD B2 :4.8F
------------------------------------2006
EC80 13 C3 B9 EB 7E 12 23 13 :3.40
EC88 7E C9 E1 ED 4B 06 EA 2A :4.7A
EC90 02 EA CD 88 19 00 13 DD :3.4A
EC98 21 00 00 E5 CD 3F 19 28 :2.53
ECA0 0F CD C7 EC 38 0A E3 23 :3.D7
ECA8 CD BA EC DD 23 E1 18 EB :5.57
ECB0 DD 22 00 EE CD 24 ED C3 :4.8E
ECB8 00 EA 23 71 23 70 23 E5 :3.19
------------------------------------1E2C
ECC0 2A 08 EA 09 E3 C1 C9 E5 :4.77
ECC8 2A 04 EA CD 45 18 E1 C9 :3.EC
ECD0 11 F8 EC CD 70 ED 21 02 :4.42
ECD8 EA CD E6 EC D8 11 0E ED :5.6D
ECE0 CD 70 ED 21 04 EA 11 0D :3.57
ECE8 4D CD C5 05 E5 EB CD A7 :5.28
ECF0 18 F7 EC E1 73 23 72 C9 :4.AD
ECF8 2A 53 54 41 52 54 20 4C :2.24
------------------------------------2162
ED00 49 4E 45 23 28 44 65 66 :2.36
ED08 3A 20 30 29 3F 0D 2A 45 :1.6E
ED10 4E 44 20 4C 49 4E 45 23 :1.FD
ED18 20 28 44 65 66 3A 6D 61 :2.5F
ED20 78 29 3F 0D 06 05 CD 14 :1.D9
ED28 0F 10 FB 2A 00 EE 11 02 :2.45
ED30 EE D5 CD D4 18 D1 CD B6 :5.D0
ED38 05 11 40 ED CD 89 08 C9 :3.6A
------------------------------------1558
ED40 20 4C 49 4E 45 53 2E 2D :1.F6
ED48 2D 3C 45 4E 44 3E 0D 11 :1.9C
ED50 77 ED 18 1C 11 84 ED 18 :3.32
ED58 17 11 91 ED 18 12 11 9C :2.7D
ED60 ED 18 0D 11 A7 ED 18 08 :2.D7
ED68 11 B0 ED 18 03 11 B6 ED :3.7D
ED70 CD B6 05 CD 29 0A C9 2A :3.7B
ED78 2A 2A 28 4D 45 4E 55 29 :1.DA
------------------------------------14EA
ED80 2A 2A 2A 0D 31 2E 3C 52 :1.78
ED88 45 4E 55 4D 42 45 52 3E :2.4C
ED90 0D 32 2E 3C 44 45 4C 45 :1.C3
ED98 54 45 3E 0D 33 2E 3C 41 :1.C2
EDA0 50 50 45 4E 44 3E 0D 34 :1.F6
EDA8 2E 3C 53 4F 52 54 3E 0D :1.FD
EDB0 35 2E 45 4E 44 0D 23 C5 :2.2F
EDB8 DD CA DE DD 3F 0D 2A 4E :4.26
------------------------------------1191
EDC0 45 57 20 4E 55 4D 42 45 :2.33
EDC8 52 20 28 44 65 66 3A 31 :2.14
EDD0 30 29 3F 0D 2A 53 54 45 :1.BB
EDD8 50 20 4E 55 4D 42 45 52 :2.39
EDE0 28 44 65 66 3A 31 30 29 :1.FB
EDE8 3F 0D 00 00 00 00 00 00 :0.4C
EDF0 00 00 00 00 00 00 00 00 :0.00
EDF8 00 00 00 00 00 00 00 00 :0.00
------------------------------------0A82
EE00 00 00 00 00 00 00 00 00 :0.00
EE08 00 00 00 00 00 00 00 00 :0.00
```

リスト2

```
0000                ;*******************************
0001                ;*  RENUMBER (MZ-1Z001 V1.0C)  *
0002                ;*      1983.3.6  BY Y.O (V1.0) *
0003                ;*******************************
0004 05B6           NLMSG: EQU   05B6H
0005 0889           MSG:   EQU   0889H
0006 0F14           BELL:  EQU   0F14H
0007 0A29           NL:    EQU   0A29H
0008 05C5           GETLBR:EQU   05C5H
0009 01CB           MLOAD: EQU   01CBH
0010 1300           BASIC: EQU   1300H
0011                       ORG   EA00H
0012                ;
0013 EA00 1808      ST:    JR    ST1
```

```
EA02 0000      STLN:  DEFW 0000H       ;START LINE#
EA04 FFFF      ENDLN: DEFW FFFFH       ;END LINE#
EA06 0A00      NEWLN: DEFW 000AH       ;NEW LINE#
EA08 0A00      STEP:  DEFW 000AH       ;ADD NUMBER
EE00           W1:    EQU  EE00H       ;WORK1
EE02           WT:    EQU  EE02H       ;NUMBER TABLE
                      ;
EA0A ED7B0A4D  ST1:   LD   SP,(4D0AH)
EA0E CD4FED           CALL MS0
EA11 CD54ED           CALL MS1
EA14 CD59ED           CALL MS2
EA17 CD5EED           CALL MS3
EA1A CD63ED           CALL MS4
EA1D CD68ED           CALL MS5
                      ;***
EA20 CD6DED    GTK:   CALL MS6
EA23 110D4D           LD   DE,4D0DH
EA26 CDC505           CALL GETLBR
EA29 3A0D4D           LD   A,(4D0DH)
EA2C FE31             CP   31H
EA2E CA4AEB           JP   Z,REN
EA31 FE32             CP   32H
EA33 CA47EA           JP   Z,DEL
EA36 FE33             CP   33H
EA38 CAE9EA           JP   Z,APP
EA3B FE34             CP   34H
EA3D CA90EA           JP   Z,SOR
EA40 FE35             CP   35H
EA42 CA0013           JP   Z,1300H
EA45 18D9             JR   GTK
EA47 1193ED    DEL:   LD   DE,MD2+2
EA4A CD70ED           CALL MSOUT
EA4D CDD0EC           CALL GTLN
EA50 38F5             JR   C,DEL
EA52 7E               LD   A,(HL)      ;HL=ENDLN+1
EA53 2B               DEC  HL
EA54 B6               OR   (HL)
EA55 200C             JR   NZ,DEL1
EA57 3EFF             LD   A,FFH
EA59 CD921A           CALL 1A92H       ;(ENDLN)=FFFFH
EA5C 2A02EA           LD   HL,(STLN)
EA5F 7C               LD   A,H
EA60 B5               OR   L
EA61 28E4             JR   Z,DEL
EA63 2100EE    DEL1:  LD   HL,W1       ;DEL COUNTER
EA66 CD911A           CALL 1A91H
EA69 2A02EA    DEL2:  LD   HL,(STLN)
EA6C CD8819           CALL 1988H
EA6F 8AEA             DEFW DEND
EA71 E5               PUSH HL
EA72 CD3F19           CALL 193FH
EA75 2A04EA           LD   HL,(ENDLN)
EA78 CD4518           CALL 1845H       ;HL<DE?
EA7B E1               POP  HL
EA7C 380C             JR   C,DEND
EA7E CDA813           CALL 13A8H
EA81 2A00EE           LD   HL,(W1)
EA84 23               INC  HL
EA85 2200EE           LD   (W1),HL
EA88 18DF             JR   DEL2
EA8A CD24ED    DEND:  CALL ENDM
EA8D C300EA           JP   ST
                      ;
EA90 11A9ED    SOR:   LD   DE,MD4+2
EA93 CD70ED           CALL MSOUT
EA96 2100EE           LD   HL,W1
EA99 CD911A           CALL 1A91H       ;W1=COUNTER
EA9C 215C54    SOR1:  LD   HL,545CH
EA9F CD3F19    SOR2:  CALL 193FH
EAA2 283F             JR   Z,SEND
EAA4 E5               PUSH HL
EAA5 D5               PUSH DE
EAA6 CD3F19           CALL 193FH
EAA9 2836             JR   Z,SENDP
EAAB E1               POP  HL
EAAC EB               EX   DE,HL
EAAD CD4518           CALL 1845H       ;FLN>B-LN?
EAB0 E1               POP  HL
EAB1 30EC             JR   NC,SOR2
EAB3 CDCA19           CALL 19CAH       ;BC=LINE LENGTH
EAB6 E5               PUSH HL
EAB7 110E4E           LD   DE,4E0EH
EABA EDB0             LDIR
EABC E1               POP  HL
EABD CDA813           CALL 13A8H       ;ラインDEL
EAC0 2A104E           LD   HL,(4E10H)
EAC3 CD8819           CALL 1988H
EAC6 00EA             DEFW ST
EAC8 220E4E           LD   (4E0EH),HL
EACB EB               EX   DE,HL
EACC 210E4E           LD   HL,4E0EH
EACF CDCA19           CALL 19CAH
EAD2 CD7A19           CALL 197AH
EAD5 CDB213           CALL 13B2H
EAD8 2A00EE           LD   HL,(W1)
EADB 23               INC  HL
EADC 2200EE           LD   (W1),HL
EADF 18BB             JR   SOR1
EAE1 F1        SENDP: POP  AF
EAE2 F1               POP  AF
EAE3 CD24ED    SEND:  CALL ENDM
EAE6 C300EA           JP   ST
EAE9 119EED    APP:   LD   DE,MD3+2
EAEC CD70ED           CALL MSOUT
EAEF CDEF1B           CALL 1BEFH       ;PT+LINE L.
EAF2 3ED8             LD   A,D8H       ;RET C
EAF4 32D901           LD   (01D9H),A
EAF7 32DA01           LD   (01DAH),A
EAFA 3EAF             LD   A,0AFH      ;XOR A
EAFC 32ED01           LD   (01EDH),A
EAFF 3EC9             LD   A,C9H
EB01 32EE01           LD   (01EEH),A
EB04 CDCB01           CALL MLOAD
                      ;
EB07 3E38             LD   A,38H
EB09 32D901           LD   (01D9H),A
EB0C 3EE2             LD   A,E2H
EB0E 32DA01           LD   (01DAH),A
EB11 3ECD             LD   A,CDH
EB13 32ED01           LD   (01EDH),A
EB16 3EB2             LD   A,B2H
EB18 32EE01           LD   (01EEH),A
EB1B 381A             JR   C,AEND
                      ;
EB1D 2A5211           LD   HL,(1152H)
EB20 2B               DEC  HL
EB21 2B               DEC  HL
EB22 4D               LD   C,L
EB23 44               LD   B,H
EB24 ED5B4E52         LD   DE,(524EH)
EB28 1B               DEC  DE
EB29 1B               DEC  DE
EB2A CD6019           CALL 1960H
EB2D CD9A1A           CALL 1A9AH
EB30 ED535411         LD   (1154H),DE
EB34 CDB202           CALL 02B2H       ;RDDAT
                      ;
EB37 CD9613    AEND:  CALL 1B96H
EB3A 0605             LD   B,5
EB3C CD140F    AE1:   CALL BELL
EB3F 10FB             DJNZ AE1
EB41 1149ED           LD   DE,MLI+9
EB44 CD70ED           CALL MSOUT
EB47 C300EA           JP   ST
EB4A 1186ED    REN:   LD   DE,MD1+2
EB4D CD70ED           CALL MSOUT
EB50 CDD0EC           CALL GTLN
EB53 38F5             JR   C,REN
EB55 7E               LD   A,(HL)
EB56 2B               DEC  HL
EB57 B6               OR   (HL)
EB58 2005             JR   NZ,RENA
EB5A 3EFF             LD   A,FFH
EB5C CD921A           CALL 1A92H       ;(ENDLN)=FFFFH
                      ;
EB5F 11BEED    RENA:  LD   DE,MR1
EB62 CD70ED           CALL MSOUT
EB65 2106EA           LD   HL,NEWLN
EB68 CDE6EC           CALL GTLNS
EB6B 38F2             JR   C,RENA
EB6D 7E               LD   A,(HL)
EB6E 2B               DEC  HL
EB6F B6               OR   (HL)
EB70 2002             JR   NZ,RENB
EB72 360A             LD   (HL),0AH
                      ;
EB74 11D4ED    RENB:  LD   DE,MR2
EB77 CD70ED           CALL MSOUT
EB7A 2108EA           LD   HL,STEP
EB7D CDE6EC           CALL GTLNS
EB80 38F2             JR   C,RENB
EB82 7E               LD   A,(HL)
EB83 2B               DEC  HL
EB84 B6               OR   (HL)
EB85 2002             JR   NZ,RENC
EB87 360A             LD   (HL),0AH
                      ;*** WT TABLE SAKUSEI ***
EB89 ED4B06EA  RENC:  LD   BC,(NEWLN)  ;
EB8D 2102EE           LD   HL,WT
EB90 E5               PUSH HL
                      ;
EB91 2A02EA           LD   HL,(STLN)
EB94 CD8819           CALL 1988H       ;LINE SEARCH
EB97 0013             DEFW BASIC       ;ERROR RET.
                      ;
EB99 CD3F19    R1:    CALL 193FH       ;HL=pt. DE=ln.
EB9C 280F             JR   Z,R2        ;pt.=0
EB9E CDC7EC           CALL ENDLC       ;ENDLN CHECK
EBA1 380A             JR   C,R2
                      ;
EBA3 E3               EX   (SP),HL
EBA4 73               LD   (HL),E
EBA5 23               INC  HL
EBA6 72               LD   (HL),D
EBA7 CDBAEC           CALL NLSET       ;BC(NEW#) SET
EBAA E3               EX   (SP),HL
EBAB 18EC             JR   R1
                      ;
```

```
EBAD E1          R2:     POP  HL
EBAE 3EFF                LD   A,FFH        ;TABLE END-MARK
EBB0 CD921A              CALL 1A92H        ;FFFFH SET
                 ;
                 ;*** UPDATE TEXT ***
EBB3 215C54              LD   HL,545CH     ;TEXT START
EBB6 E5                  PUSH HL
EBB7 1805                JR   R3
EBB9 E1          RNXT:   POP  HL
EBBA CD3F19              CALL 193FH        ;HL=pt. DE=ln.
EBBD E5                  PUSH HL
EBBE E5          R3:     PUSH HL
EBBF CD3F19              CALL 193FH        ;HL=pt. DE=ln.
EBC2 E1                  POP  HL
EBC3 CA8AEC              JP   Z,LSET
                 ;
EBC6 110E4E              LD   DE,4E0EH
EBC9 CD84EC              CALL INCHL
EBCC CD84EC              CALL INCHL
EBCF CD84EC              CALL INCHL
EBD2 CD84EC      R4:     CALL INCHL
EBD5 7E          R5:     LD   A,(HL)
EBD6 FE0D                CP   0DH          ;LINE END?
EBD8 CA61EC              JP   Z,RLE
EBDB FE22                CP   22H          ;(")?
EBDD 2830                JR   Z,C22H
                 ;
EBDF FE80                CP   80H          ;STATEMENT
EBE1 20EF                JR   NZ,R4
EBE3 CD84EC              CALL INCHL
EBE6 FE86                CP   86H          ;RUN?
EBE8 2832                JR   Z,SCN
EBEA FE8C                CP   8CH          ;THEN?
EBEC 282E                JR   Z,SCN
EBEE FE8D                CP   8DH          ;GOTO?
EBF0 282A                JR   Z,SCN
EBF2 FE8E                CP   8EH          ;GOSUB?
EBF4 2826                JR   Z,SCN
EBF6 FE9C                CP   9CH          ;RESTORE?
EBF8 2822                JR   Z,SCN
                 ;
EBFA FE80                CP   80H          ;REM?
EBFC 2804                JR   Z,SK
EBFE FE81                CP   81H          ;DATA?
EC00 20D0                JR   NZ,R4
                 ;
EC02 CD84EC      SK:     CALL INCHL        ;SKIP 0DH,":"
EC05 FE0D                CP   0DH
EC07 2858                JR   Z,RLE
EC09 FE3A                CP   3AH          ;":"
EC0B 28C5                JR   Z,R4
EC0D 18F3                JR   SK
                 ;
EC0F CD84EC      C22H:   CALL INCHL
EC12 FE0D                CP   0DH
EC14 284B                JR   Z,RLE
EC16 FE22                CP   22H
EC18 20F5                JR   NZ,C22H      ;YOMITOBASHI
EC1A 18B6                JR   R4
                 ;*** LINE# CHECK ***
EC1C CD84EC      SCN:    CALL INCHL
EC1F D5          SCN1:   PUSH DE
EC20 CDA718              CALL 18A7H        ;DEC(HL)→BIN(DE)
EC23 44EC                DEFW RS4
EC25 7A                  LD   A,D
EC26 B3                  OR   E
EC27 281B                JR   Z,RS4
EC29 2200EE              LD   (W1),HL
EC2C D5                  PUSH DE
EC2D 2102EE              LD   HL,WT
EC30 7E          RS1:    LD   A,(HL)
EC31 4F                  LD   C,A
EC32 23                  INC  HL
EC33 A6                  AND  (HL)         ;(HL)=FFFFH?
EC34 3C                  INC  A
EC35 2816                JR   Z,RNSET
                 ;
EC37 79                  LD   A,C
EC38 BB                  CP   E
EC39 2004                JR   NZ,RS3
EC3B 7E                  LD   A,(HL)
EC3C BA                  CP   D
EC3D 2808                JR   Z,RP
EC3F 23          RS3:    INC  HL
EC40 23                  INC  HL
EC41 23                  INC  HL
EC42 18EC                JR   RS1
EC44 D1          RS4:    POP  DE
EC45 188E                JR   R5
                 ;
EC47 23          RP:     INC  HL
EC48 5E                  LD   E,(HL)
EC49 23                  INC  HL
EC4A 56                  LD   D,(HL)
EC4B EB                  EX   DE,HL
EC4C E3                  EX   (SP),HL
EC4D E1          RNSET:  POP  HL
EC4E D1                  POP  DE
EC4F CDD418              CALL 18D4H        ;BIN→DEC
                 ;
EC52 2A00EE              LD   HL,(W1)
EC55 CD6718              CALL 1867H
EC58 2C                  DEFB 2CH          ;","
EC59 D5EB                DEFW R5           ;NOT ","
EC5B 3E2C                LD   A,2CH        ;","
EC5D 12                  LD   (DE),A
EC5E 13                  INC  DE
EC5F 18BE                JR   SCN1
                 ;
EC61 CD84EC      RLE:    CALL INCHL
EC64 CDED14              CALL 14EDH
EC67 CDE814              CALL 14E8H
EC6A E1                  POP  HL
EC6B E5                  PUSH HL
EC6C CDA813              CALL 13A8H
EC6F E1                  POP  HL
EC70 E5                  PUSH HL
EC71 220E4E              LD   (4E0EH),HL
EC74 EB                  EX   DE,HL
EC75 210E4E              LD   HL,4E0EH
EC78 CDCA19              CALL 19CAH
EC7B CD7A19              CALL 197AH
EC7E CDB213              CALL 13B2H
EC81 C3B9EB              JP   RNXT
EC84 7E          INCHL:  LD   A,(HL)
EC85 12                  LD   (DE),A
EC86 23                  INC  HL
EC87 13                  INC  DE
EC88 7E                  LD   A,(HL)
EC89 C9                  RET
                 ;
                 ;*** LINE# SET ***
EC8A E1          LSET:   POP  HL
EC8B ED4B06EA            LD   BC,(NEWLN)
EC8F 2A02EA              LD   HL,(STLN)
EC92 CD8819              CALL 1988H
EC95 0013                DEFW BASIC
EC97 DD210000            LD   IX,0         ;LINE COUNT
EC9B E5          LS1:    PUSH HL
EC9C CD3F19              CALL 193FH
EC9F 280F                JR   Z,REND
ECA1 CDC7EC              CALL ENDLC
ECA4 380A                JR   C,REND
                 ;
ECA6 E3                  EX   (SP),HL
ECA7 23                  INC  HL
ECA8 CDBAEC              CALL NLSET
ECAB DD23                INC  IX
ECAD E1                  POP  HL
ECAE 18EB                JR   LS1
                 ;
ECB0 DD2200EE    REND:   LD   (W1),IX
ECB4 CD24ED              CALL ENDM
ECB7 C300EA              JP   ST
                 ;
                 ;*** SUBROUTINE ***
                 ;*** NEW LINE# SET ***
ECBA 23          NLSET:  INC  HL
ECBB 71                  LD   (HL),C
ECBC 23                  INC  HL
ECBD 70                  LD   (HL),B
ECBE 23                  INC  HL
ECBF E5                  PUSH HL
ECC0 2A08EA              LD   HL,(STEP)
ECC3 09                  ADD  HL,BC
ECC4 E3                  EX   (SP),HL
ECC5 C1                  POP  BC
ECC6 C9                  RET
                 ;
                 ;*** LINE END CHECK ***
ECC7 E5          ENDLC:  PUSH HL
ECC8 2A04EA              LD   HL,(ENDLN)
ECCB CD4518              CALL 1845H
ECCE E1                  POP  HL
ECCF C9                  RET
                 ;
                 ;*** SUBROUTINE ***
ECD0 11F8EC      GTLN:   LD   DE,MST
ECD3 CD70ED              CALL MSOUT
ECD6 2102EA              LD   HL,STLN
ECD9 CDE6EC              CALL GTLNS
ECDC D8                  RET  C
ECDD 110EED              LD   DE,MEN
ECE0 CD70ED              CALL MSOUT
ECE3 2104EA              LD   HL,ENDLN
                 ;
ECE6 110D4D      GTLNS:  LD   DE,4D0DH
ECE9 CDC505              CALL GETLBR
ECEC E5                  PUSH HL
ECED EB                  EX   DE,HL
ECEE CDA718              CALL 18A7H
ECF1 F7EC                DEFW GTLRE
ECF3 E1                  POP  HL
ECF4 73                  LD   (HL),E
ECF5 23                  INC  HL
ECF6 72                  LD   (HL),D
ECF7 C9          GTLRE:  RET               ;CF=0:OK,CF=1:NG
                 ;
```

```
ECF8 2A535441 MST:    DEFM '*START LINE#(Def: 0)?'
ECFC 5254204C
ED00 494E4523
ED04 28446566
ED08 3A203029
ED0C 3F
ED0D 0D               DEFB 0DH
ED0E 2A454E44 MEN:    DEFM '*END LINE# (Def:max)?'
ED12 204C494E
ED16 45232028
ED1A 4465663A
ED1E 6D617829
ED22 3F
ED23 0D               DEFB 0DH
                ;
ED24 0605     ENDM:   LD   B,5
ED26 CD140F   ENDM1:  CALL BELL
ED29 10FB             DJNZ ENDM1
ED2B 2A00EE           LD   HL,(W1)
ED2E 1102EE           LD   DE,WT
ED31 D5               PUSH DE
ED32 CDD418           CALL 18D4H           ;BIN→DEC
ED35 D1               POP  DE
ED36 CDB605           CALL NLMSG
ED39 1140ED           LD   DE,MLI
ED3C CD8908           CALL MSG
ED3F C9               RET
                ;
ED40 204C494E MLI:    DEFM ' LINES.--<END>'
ED44 45532E2D
ED48 2D3C454E
ED4C 443E
ED4E 0D               DEFB 0DH
ED4F 1177ED   MS0:    LD   DE,MD0
ED52 181C             JR   MSOUT
ED54 1184ED   MS1:    LD   DE,MD1
ED57 1817             JR   MSOUT
ED59 1191ED   MS2:    LD   DE,MD2
ED5C 1812             JR   MSOUT
ED5E 119CED   MS3:    LD   DE,MD3
ED61 180D             JR   MSOUT
ED63 11A7ED   MS4:    LD   DE,MD4
ED66 1808             JR   MSOUT
ED68 11B0ED   MS5:    LD   DE,MD5
ED6B 1803             JR   MSOUT
ED6D 11B6ED   MS6:    LD   DE,MD6
ED70 CDB605   MSOUT:  CALL NLMSG
ED73 CD290A           CALL NL
ED76 C9               RET
                ;
ED77 2A2A2A28 MD0:    DEFM '***(MENU)***'
ED7B 4D454E55
ED7F 292A2A2A
ED83 0D               DEFB 0DH
ED84 312E3C52 MD1:    DEFM '1.<RENUMBER>'
ED88 454E554D
ED8C 4245523E
ED90 0D               DEFB 0DH
ED91 322E3C44 MD2:    DEFM '2.<DELETE>'
ED95 454C4554
ED99 453E
ED9B 0D               DEFB 0DH
ED9C 332E3C41 MD3:    DEFM '3.<APPEND>'
EDA0 5050454E
EDA4 443E
EDA6 0D               DEFB 0DH
EDA7 342E3C53 MD4:    DEFM '4.<SORT>'
EDAB 4F52543E
EDAF 0D               DEFB 0DH
EDB0 352E454E MD5:    DEFM '5.END
EDB4 44
EDB5 0D               DEFB 0DH
EDB6 23C5DDCA MD6:    DEFM '#ﾅﾝﾊﾞﾝ?'
EDBA DEDD3F
EDBD 0D               DEFB 0DH
EDBE 2A4E4557 MR1:    DEFM '*NEW NUMBER (Def:10)?'
EDC2 204E554D
EDC6 42455220
EDCA 28446566
EDCE 3A313029
EDD2 3F
EDD3 0D               DEFB 0DH
EDD4 2A535445 MR2:    DEFM '*STEP NUMBER(Def:10)?'
EDD8 50204E55
EDDC 4D424552
EDE0 28446566
EDE4 3A313029
EDE8 3F
EDE9 0D               DEFB 0DH
                ;*** END ***
```

MZ-2200/2000/80B用

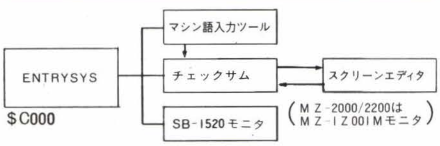
マシン語入力ツール——強化版ENTRYSYS

西川 幹雄

はじめに

Oh! MZ 6月号の「マシン語特集」に刺激されて，3月号「MZ-2000マシン語入力ツールの作成」と6月号「チェックサムプログラム」を合体させ，さらに「エディタ機能」を付加したプログラムを作ってみました。これによりマシン語プログラムの移植が，よりかんたんにできるというものです。名付けて「ENTRYSYS」。

ENTRYSYSの構成



プログラム内容

1.〝HEX DATA ENTRY″

3月号の「MZ-2000マシン語入力ツールの作成」と同じ操作法になっています。ただし，右手だけでキーインできるよう，次のように変換しました。

[+]→[A]　[−]→[B]　[.]→[C]
[00]→[D]　[ ]]→[E]　[?]→[F]

[INST] キーでメインメニューに戻ります。[BREAK] キーで〝CHECKSUM″に移ります。

このとき，キー配列はノーマルに戻っています。

2.〝CHECKSUM″

6月号の「チェックサム」と同じ操作法です。ただし，[SPACE] キーを1回押すとSTOPし，さらに [SPACE] キーを押すと処理を続けます。

[HOME] キーでチェックサム開始アドレスを変更できます。

[INST] キーでメインメニューに戻ります。

[BREAK] キーで〝SCREDIT″に移ります。

3.〝SCREDIT″

・チェックサムでキー入力ミスを発見した場合は，[SPACE] キーでSTOPさせ，さらに [BREAK] キーで〝SCREDIT″に制御を移します。そうすると，アドレスに続いてデータの位置にカーソルが点滅しますのでカーソルキーで移動して正しいデータを入力してください。1行の最後で [CR] キーを押すと，メモリに書かれます。正しくメモリに書かれると，カーソルは次の行に移ります。

・0〜9，A〜B以外のキーを入力すると，ベルが鳴ってカーソルがその行のデータの先頭に戻るようになっています。

・〝?″を使うと，キャラクタのままの入力が可能なので，アスキーコードを調べずに済むようになっていて便利です。

例）?A──→$41と入力される。

・〝M″を入力するとメインメニューに戻ります。

・[BREAK] キーを押すと〝CHECKSUM″に戻るので，再チェックして最初の処理が続行できます。

図1 ENTRYSYSメニュー

- メインメニュー
  - HEX DATA ENTRY
    - MONITOR RETURN
    - ADR. BACK
    - CHANGE ADR.
    - NEXT ADR.
    - go to MENU（メインメニュー）
    - go to CHECKSUM
  - CHECKSUM
    - PRINT OUT
    - CHANGE ADR.
    - STOP
    - go to MENU（メインメニュー）
    - go to SCREDIT
      - ASCII入力
      - WRITE MEM
      - go to MENU（メインメニュー）
      - go to CHECKSUM
  - MONITOR

注：操作はこの図のように行なってください。

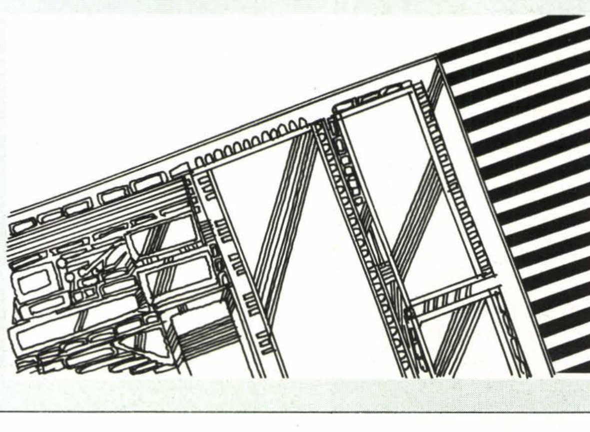
リスト

```
** Oh!MZ Check sum system V1.0 **

ADR. +0 +1 +2 +3 +4 +5 +6 +7 :sum.
C000 CD EE 0C 31 40 11 CD 87 :3.9D
C008 C0 CD 91 C0 CD 32 08 FE :4.E3
C010 00 28 F9 FE 31 CC CE C3 :4.AD
C018 FE 32 CC 47 C1 FE 39 CC :5.07
C020 2D C0 CC 7E C0 CA B1 00 :4.72
C028 CD 14 0F 18 DF 21 0B 00 :2.13
C030 36 01 21 0C 00 36 18 CD :1.7F
C038 D3 08 C9 21 0B 00 36 07 :2.0D
-----------------------------------1C45
C040 21 0C 00 36 18 CD D3 08 :2.23
C048 21 00 05 22 D1 11 11 55 :1.90
C050 C0 CD 89 08 C9 90 90 90 :4.97
C058 90 90 90 90 90 90 90 90 :4.80
C060 90 90 90 90 90 90 90 90 :4.80
C068 90 90 90 90 90 90 90 90 :4.80
C070 90 90 90 90 90 90 90 90 :4.80
C078 90 90 90 90 90 0D 11 85 :3.73
-----------------------------------1DBD
C080 C0 CD 89 08 C9 06 0D CD :3.C7
C088 2D C0 CD 7E C0 CD 3B C0 :4.C0
C090 C9 21 02 02 22 D1 11 11 :2.03
C098 CE C0 CD 89 08 21 09 08 :3.1E
C0A0 22 D1 11 11 F4 C0 CD 89 :4.1F
C0A8 08 21 09 0B 22 D1 11 11 :1.52
C0B0 0B C1 CD 89 08 21 09 10 :2.64
C0B8 22 D1 11 11 22 C1 CD 89 :3.4E
-----------------------------------18CB
C0C0 08 21 0C 15 22 D1 11 11 :1.5F
C0C8 39 C1 CD 89 08 C9 2A 2A :3.75
C0D0 20 20 48 45 58 20 44 41 :1.CA
C0D8 54 41 20 45 4E 54 52 59 :2.47
C0E0 20 53 55 50 50 4F 52 54 :2.5D
C0E8 20 53 59 53 54 45 4D 20 :2.25
C0F0 20 2A 2A 0D 48 45 58 20 :1.86
C0F8 44 41 54 41 20 45 4E 54 :2.21
-----------------------------------110E
C100 52 59 20 2D 2D 2D 2D 2D :1.AC
C108 20 31 0D 43 48 45 43 4B :1.BC
C110 53 55 4D 20 20 20 20 20 :1.95
C118 20 20 2D 2D 2D 2D 2D 20 :1.41
C120 32 0D 4D 4F 4E 49 54 4F :2.15
C128 52 20 72 65 74 75 72 6E :3.12
C130 20 2D 2D 2D 2D 2D 20 39 :1.5A
C138 0D 48 20 49 20 54 20 20 :1.72
-----------------------------------0E31
C140 20 31 20 7E 20 39 0D CD :2.22
C148 87 C0 21 00 00 22 D1 11 :2.6C
C150 11 52 C3 CD 89 08 21 00 :2.A5
C158 05 22 D1 11 08 AF 08 CD :2.95
C160 DE C2 7F 20 20 2A 2A 20 :2.D3
C168 4F 68 21 4D 5A 20 43 68 :2.4A
C170 65 63 6B 20 73 75 6D 20 :2.C8
C178 56 31 2E 30 20 2A 2A 7F :1.D8
-----------------------------------1385
C180 00 CD DE C2 53 74 61 72 :4.07
C188 74 20 41 64 72 2E 24 3D :2.3A
C190 00 11 AB 10 CD A4 06 1A :2.5D
C198 FE 0B CC 14 0F 28 BD 11 :2.EE
C1A0 B7 10 1A FE 3F 20 05 08 :2.4B
C1A8 3E 01 08 13 CD 14 06 38 :1.79
C1B0 AB E5 2A D1 11 E5 21 00 :3.A2
C1B8 00 22 D1 11 11 63 C3 CD :3.08
-----------------------------------15FA
C1C0 89 08 21 00 02 22 D1 11 :1.B8
C1C8 11 88 C3 CD 89 08 21 14 :2.EF
C1D0 02 22 D1 11 11 99 C3 CD :3.40
C1D8 89 08 21 00 04 22 D1 11 :1.BA
C1E0 11 AA C3 CD 89 08 E1 22 :3.DF
C1E8 D1 11 E1 CD DE C2 2A 2A :4.84
C1F0 20 4F 68 21 4D 5A 20 43 :2.02
C1F8 68 65 63 6B 20 73 75 6D :3.10
-----------------------------------1716
C200 20 73 79 73 74 65 6D 20 :2.E5
C208 56 31 2E 30 20 2A 2A 7F :1.D8
C210 7F 00 CD DE C2 41 44 52 :3.C3
C218 2E 20 2B 30 20 2B 31 20 :1.45
C220 2B 32 20 2B 33 20 2B 34 :1.5A
C228 20 2B 35 20 2B 36 20 2B :1.4C
C230 37 20 3A 73 75 6D 2E 7F :2.93
C238 00 0E 08 D9 01 00 00 D9 :1.C9
-----------------------------------10C7
C240 E5 CD C0 C2 CD DA C2 E1 :6.7E
C248 06 08 D9 21 00 00 D9 7E :2.5F
C250 23 D9 16 00 5F 19 D9 CD :3.30
C258 C5 C2 CD DA C2 10 F0 3E :5.2E
C260 3A CD 01 C3 D9 7C D9 CD :4.C6
C268 F3 05 CD 01 C3 3E 2E CD :3.C2
C270 01 C3 D9 7D D9 CD C5 C2 :5.47
C278 CD F4 C2 D9 09 E5 C1 D9 :5.E4
-----------------------------------24EE
C280 0D 20 15 E5 D9 C5 D9 06 :3.A4
C288 22 3E 2D CD 01 C3 10 F9 :3.27
C290 E1 CD C0 C2 CD F4 C2 E1 :6.94
C298 E5 AF CD 32 08 E1 FE 20 :4.9A
C2A0 20 07 CD 32 08 FE 00 28 :2.54
C2A8 F9 FE 0B CC BB C5 FE 05 :5.51
C2B0 CA 47 C1 FE 08 CA 03 C0 :4.65
C2B8 79 B7 C2 40 C2 C3 39 C2 :4.B2
-----------------------------------22B5
C2C0 7C CD C5 C2 7D F5 E6 F0 :6.18
C2C8 0F 0F 0F 0F CD F3 05 CD :2.CE
C2D0 01 C3 F1 E6 0F CD F3 05 :4.6F
C2D8 18 27 3E 20 18 23 E3 7E :2.39
C2E0 23 B7 28 09 FE 7F 28 07 :2.B7
C2E8 CD 01 C3 18 F2 E3 C9 CD :5.14
C2F0 F4 C2 18 EB 3E 0D CD 01 :3.D2
C2F8 C3 3E 0A 08 B7 28 0B 18 :2.15
-----------------------------------1D40
C300 0B F5 CD C6 08 F1 08 B7 :4.4B
C308 20 02 08 C9 08 F5 3E 00 :2.2E
C310 CD 23 C3 F1 D3 FF 3E 80 :5.34
C318 D3 FE 3E 01 CD 23 C3 AF :4.72
C320 D3 FE C9 C5 D5 57 1E 0C :4.B5
C328 01 00 00 DB FE E6 0D BA :3.87
C330 20 03 D1 C1 C9 0B 78 B1 :3.B2
C338 20 F1 1D 20 EE D1 C1 08 :3.D6
-----------------------------------1FE3
C340 AF 08 CD DE C2 7F 45 52 :4.3A
C348 52 4F 52 20 21 7F 00 C3 :2.76
C350 5C C1 3F 20 2D 2D 2D 50 :2.53
C358 52 49 4E 54 45 52 20 4F :2.43
C360 55 54 0D 20 20 20 20 20 :1.56
C368 20 20 20 20 20 20 20 20 :1.00
C370 20 20 20 20 20 20 20 53 :1.33
C378 50 41 43 45 20 6B 65 79 :2.82
-----------------------------------1151
C380 2D 2D 2D 53 54 4F 50 0D :1.DA
C388 48 4F 4D 45 2D 2D 43 48 :2.0E
C390 41 4E 47 20 41 44 52 2E :1.FB
C398 0D 49 4E 53 54 2D 2D 2D :1.D2
```

```
C3A0 67 6F 74 6F 20 4D 45 4E :2.B9
C3A8 55 0D 42 52 45 41 4B 20 :1.E7
C3B0 6B 65 79 20 2D 2D 2D 20 :2.10
C3B8 67 6F 20 74 6F 20 3C 53 :2.88
-----------------------------------10ED
C3C0 43 52 45 44 49 54 3E 20 :2.19
C3C8 50 52 4F 47 2E 0D CD 87 :2.C7
C3D0 C0 21 00 00 22 D1 11 11 :1.F6
C3D8 FE C4 CD 89 08 21 00 02 :3.43
C3E0 22 D1 11 11 93 C5 CD 89 :3.C3
C3E8 08 3E 41 32 2C 0D 3C 32 :1.60
C3F0 2D 0D 3C 32 2B 0D 3C 32 :1.4E
C3F8 2A 0D 3C 32 6E 0D 3C 32 :1.8E
-----------------------------------1218
C400 5B 0D 21 00 05 22 D1 11 :1.92
C408 3E 4D CD 8C 05 E5 2A D1 :3.C9
C410 11 E5 21 00 02 22 D1 11 :2.1D
C418 11 23 C5 CD 89 08 21 14 :2.8C
C420 02 22 D1 11 11 38 C5 CD :2.E1
C428 89 08 21 00 03 22 D1 11 :1.B9
C430 11 49 C5 CD 89 08 21 14 :2.B2
C438 03 22 D1 11 11 5A C5 CD :3.04
-----------------------------------1454
C440 89 08 21 00 04 22 D1 11 :1.BA
C448 11 6B C5 CD 89 08 E1 22 :3.A2
C450 D1 11 E1 0E 01 CD BC 05 :3.60
C458 7E CD DD 05 CD C4 08 CD :4.93
C460 14 0F CD 32 08 FE 05 28 :2.55
C468 9F FE 0B CA CC C4 FE 07 :5.07
C470 28 4D FE 0D 28 19 FE 08 :2.C7
C478 28 4D CD C6 08 CD FD 05 :3.DF
-----------------------------------1B51
C480 38 E0 53 5F 0D 28 D8 00 :2.D7
C488 7A 07 07 07 07 B3 77 3E :1.FE
C490 07 57 A5 23 AA 20 BC 2B :2.D7
C498 7D 92 6F 0E 01 06 08 CD :2.68
C4A0 C4 08 3E 3A CD C6 08 AF :3.8E
C4A8 86 23 10 FC CD DD 05 0D :3.71
C4B0 20 A1 CB 5D 20 9D 2B 3E :3.0F
C4B8 F0 A5 6F 06 10 18 E0 79 :3.8B
-----------------------------------17AD
C4C0 B7 28 90 2B 18 8D 00 21 :2.60
C4C8 C6 C4 36 01 3E 2B 32 2C :2.88
C4D0 0D 3E 2D 32 2D 0D 3E 2E :1.50
C4D8 32 2B 0D 3E 1A 32 2A 0D :1.2B
C4E0 3E 5D 32 6E 0D 3E 3F 32 :1.F7
C4E8 5B 0D 21 C6 C4 7E FE 01 :3.90
C4F0 3E 00 77 CC 87 C0 CC 91 :4.25
C4F8 C0 C8 CD 47 C1 C9 2B 20 :4.71
-----------------------------------1580
C500 83 20 41 2C 2D 20 83 20 :2.00
C508 42 2C 2E 20 83 20 43 2C :1.CE
C510 30 30 20 83 20 44 2C 5D :1.F0
C518 20 83 20 45 2C 3F 20 83 :2.16
C520 20 46 0D 44 45 4C 2D 2D :1.A2
C528 2D 41 44 52 2E 42 41 43 :1.F8
C530 4B 20 20 20 20 20 20 0D :1.18
C538 48 4F 4D 45 2D 2D 43 48 :2.0E
-----------------------------------0E94
C540 41 4E 47 20 41 44 52 2E :1.FB
C548 0D 43 52 6B 65 79 2D 2D :2.45
C550 4E 45 58 54 20 41 44 52 :2.36
C558 2E 0D 49 4E 53 54 2D 2D :1.D3
C560 2D 67 6F 74 6F 20 4D 45 :2.98
C568 4E 55 0D 42 52 45 41 4B :2.15
C570 20 6B 65 79 20 2D 2D 2D :2.10
C578 67 6F 20 74 6F 20 3C 43 :2.78
-----------------------------------117E
C580 48 45 43 4B 53 55 4D 3E :2.4E
```

```
C588 20 50 52 4F 47 2E 20 20 :1.C6
C590 20 20 0D 42 52 45 41 4B :1.B2
C598 20 6B 65 79 20 2D 2D 2D :2.10
C5A0 20 4D 4F 4E 49 54 4F 52 :2.48
C5A8 20 72 65 74 75 72 6E 20 :2.E0
C5B0 20 20 20 20 20 20 20 20 :1.00
C5B8 20 20 0D 21 00 00 22 D1 :1.61
-----------------------------------0F5F
C5C0 11 11 5B C6 CD 89 08 21 :2.C2
C5C8 14 00 22 D1 11 11 6D C6 :2.5C
C5D0 CD 89 08 21 00 02 22 D1 :2.74
C5D8 11 11 82 C6 CD 89 08 21 :2.E9
C5E0 00 04 22 D1 11 11 AA C6 :2.89
C5E8 CD 89 08 F3 C5 0E E8 ED :4.F9
C5F0 40 CB F8 ED 41 AF 11 00 :3.F1
C5F8 D0 F5 CD 14 06 30 0F 01 :2.EC
-----------------------------------18DA
C600 28 00 EB 09 EB F1 3C FE :4.32
C608 19 CC 47 C1 18 EB F1 67 :4.48
C610 2E 06 22 D1 11 3E 50 32 :1.F8
C618 A2 06 11 93 10 3E 05 32 :1.D1
C620 D1 11 CD A4 06 1A FE 0B :3.7C
C628 CC 47 C1 CD 14 06 38 21 :3.14
C630 11 97 10 06 08 13 1A FE :1.F1
C638 3F 28 11 FE 4D CA 03 C0 :3.50
-----------------------------------1814
C640 CD 23 06 38 0C 77 23 10 :1.E4
C648 EC C3 15 C6 13 1A 13 18 :2.E2
C650 F4 CD 14 0F 21 D2 11 35 :3.1D
C658 C3 15 C6 31 73 74 20 64 :3.3A
C660 61 74 61 3D 3F 20 2D 41 :2.40
C668 53 43 49 49 0D 43 52 20 :1.EA
C670 6B 65 79 20 2D 2D 2D 20 :2.10
C678 57 52 49 54 45 20 4D 45 :2.3D
-----------------------------------1394
C680 4D 0D 31 73 74 20 64 61 :2.57
C688 74 61 3D 4D 20 2D 2D 4D :2.26
C690 45 4E 55 20 20 20 20 20 :1.88
C698 20 20 20 20 20 20 20 20 :1.00
C6A0 20 20 20 20 20 20 20 20 :1.00
C6A8 20 0D 42 52 45 41 4B 20 :1.B2
C6B0 6B 65 79 20 2D 2D 2D 20 :2.10
C6B8 67 6F 20 74 6F 20 3C 43 :2.78
-----------------------------------0E3F
C6C0 48 45 43 4B 53 55 4D 3E :2.4E
C6C8 20 50 52 4F 47 2E 0D 00 :1.93
C6D0 00 00 00 00 00 00 00 00 :0.00
C6D8 00 00 00 00 00 00 00 00 :0.00
C6E0 00 00 00 00 00 00 00 00 :0.00
C6E8 00 00 00 00 00 00 00 00 :0.00
C6F0 00 00 00 00 00 00 00 00 :0.00
C6F8 00 00 00 00 00 00 00 00 :0.00
-----------------------------------03E1
```



MZ-2200/2000/80B

PATTERN EDITOR V.1.0

吉田 匡輝

Pattern文のデータを作るプログラムです。市販ソフトもあるのですが，金銭的理由から自分で作りました。自分で使うために作ったので，マルチステートメントはほとんど使わず，またREM文を多用して作りました。このほうがBUGが少なくて済むようです。

使用法

まず，DATA文開始行を入力します。ただ CR すれば，10000行以降からになります（２回目からは続きの行番号)。

次に８×８ドットか16×16ドットかを決めます。16にすると漢字もかんたんに作れます。さらに，DATA作成かプログラム作成かを決めます。なお，終わるときは必ず９のENDで終わるようにしてください。これで，END処理をしてから終わることになります。

①DATAファイルサクセイ

- 2 4 6 8 でカーソルを動かす。
- CR キーでドットを打つ。
- SPACE バーでドットを消す。
- 5 で右上にパターンを書く。
- CLR キーで全部を消す。
- TAB キーでデータを作る。
- F キーでメディアに行番号付きシーケンシャルファイルを作る。
- E キーでメインメニューへ戻るので，１か２か９を押す。

②プログラム作成

MZではHu-BASICでないとプログラムの自己増殖ができないため，スクリーンエディットを利用しています。

データファイルは画面が25行なので，１度に20行しかできません。大量に作るときは，ファンクションキーの F 1 を押してください。RUNしても行番号が続きになるようにしてあります。

なお，データファイルのファイルネームはまちがいをなくすために“DATA１”，“DATA２”とするのが良いでしょう。

次に，プログラム作成部では行番号付きのデータ文を書いてENDしますので，極力行を節約して，NEW，次に上から順に CR，次にAPPENDしてください。テープBASICの場合は，NEW，LOAD，CR CR……とします。

APPENDあるいはLOADするプログラムがなければ，単にNEW，CR です。

プログラムについて

900～990行は，特殊キー(🅷，⇨など)を入力するためのルーチンです。ここで，「マイコン」1983年２月号を参考にしました。

プログラムは，Elder Man氏にならって，オールBASICで作ってみました。サブルーチンブロック構成で，DATA文は10000行からです。

ぼくも氏のように，オールBASICで行けるところまで行ってみようと思っています。コントロールコードの入力の方法は，127ページをご参照ください。

リスト

```
10 REM-----------------------------------------
11 REM
12 REM      Pattern Editor  Ver.1.0
13 REM
14 REM      1983.7.25       M.Yoshida
15 REM
16 REM-----------------------------------------
17 REM
18 REM
19 REM  A$(50)..................ﾃﾞｰﾀ ｶｸﾉｳ ﾖｳ
20 REM  STAT....................DATAﾌﾞﾝ ｶｲｼ ｷﾞｮｳ
21 REM  C.......................max ﾄﾞｯﾄ ｾｯﾃｲ
22 REM  B$......................GET ﾖｳ ﾍﾝｽｳ
23 REM  D$ or D.................GET ﾖｳ ﾍﾝｽｳ
24 REM  C1$.....................graphic CURSOR print
25 REM  C0$.....................graphic CURSOR ｹｼ
26 REM  F.......................DATA ﾌﾞﾝ ﾉ number 0~50
27 REM  PAT$....................PATTERN print 16 ﾖｳ
28 REM  N.......................DATA ﾌﾞﾝ ﾉ ﾅﾝﾊﾞｰ
29 REM
100 REM*** MAIN  ﾙｰﾁﾝ ***
110 CONSOLEC40,S0,24:GRAPH I1,C,O1:N=0
120 DIM A$(50)
125 GOSUB 900:REM ⇩⇧⇨⇦🅷🅲█▒● INPUT
130 STAT=10000:C=8:X=0:Y=0:XX=0:YY=0
132 C1$=STRING$(CHR$(170),8)
```

```
134 CO$=STRING$(CHR$(0),8)
136 DEF KEY(1)=CONSOLEC40!GOTO 140↲
140 PRINT"■DATAﾌﾞﾝ ｶｲｼ ｷﾞｮｳ ﾊ? ";STAT;
150 INPUT"■DATAﾌﾞﾝ ｶｲｼ ｷﾞｮｳ ﾊ?";STAT
155 IF (STAT<10000)+(STAT>65000) THEN 150
160 PRINT
170 GRAPH I1,C,O1
171 PRINT"■MAX ﾄﾞｯﾄ ﾊ? 8 or 16? ";C
172 INPUT"■MAX ﾄﾞｯﾄ ﾊ? 8 or 16?";C
175 IF (C<>8)*(C<>16) THEN 170
180 PRINT
190 PRINT"    1: DATA ﾌｧｲﾙ ｻｸｾｲ":PRINT
200 PRINT"    2: program ｻｸｾｲ"
202 PRINT
205 PRINT"    9: END"
210 GET B$:IF B$="" THEN 210
220 ON ASC(B$)-48 GOTO 1000,5000
230 IF B$="9" THEN 800
240 GOTO 180
310 X=0:XX=0:Y=0:YY=0
800 REM***  END ｼｮﾘ ***
810 CONSOLE C80,S0,24
820 GRAPH I1,C,O1
830 DEF KEY(1)=RUN  ↲
840 END
900 REM*** ■■■■■■■■■ INPUT ***
910 FOR A=0 TO 9
920 POKE4350+A,VAL(MID$("001002003004005006030031147013",A*3+1,3))
930 NEXT
940 POKE3530,28:POKE1990,48
950 FOR A=0 TO 9
960 POKE2070+A,VAL(MID$("017254016000000000000000032003",A*3+1,3))
970 NEXT
990 RETURN
1000 REM*** DATA ﾌｧｲﾙ ｻｸｾｲ ***
1010 PRINTCHR$(6);:GRAPH I1,C,O1
1020 GOSUB 2000:REM ｶﾞﾒﾝ PRINT
1025 GOSUB 2500:REM CURSOR print
1030 GET D$:IF D$="" THEN 1030
1040 D=ASC(D$)
1050 IF D=5   GOTO 3000:REM ALL CLEAR  (CLR)
1060 IF D=13 GOSUB 3500:REM DOT SET     (CR)
1070 IF D=27 GOSUB 4000:REM DATA ﾌﾞﾝ ｻｸｾｲ (TAB)
1080 IF D=32 GOSUB 3300:REM DOT RESET  (SPACE)
1090 IF (D>49)*(D<57)*(D<>51)*(D<>53)*(D<>55) GOSUB 2500:REM CURSOR move(■~■)
1100 IF D=53 GOSUB 2800:REM graphic pattern   (■)
1110 IF D=69 THEN GOTO 170:REM MAIN ﾒﾆｭｰ ﾍ  (■)
1120 IF D$="F" GOSUB 4400:REM DATA file ｻｸｾｲ (■)
1130 GOTO 1030
2000 REM*** ｶﾞﾒﾝ print ***
2010 PRINTCHR$(6);
2020 CURSOR 20, 0:PRINT"  ■
2030 CURSOR 20, 1:PRINT" ■ ■
2040 CURSOR 20, 2:PRINT"  ■
2050 CURSOR 20, 3:PRINT"■-----graphic print
2060 CURSOR 20, 4:PRINT"■■■---ALL CLEAR(DOT)";
2070 CURSOR 20, 5:PRINT"■■■■■-RESET
2080 CURSOR 20, 6:PRINT"■■----SET
2090 CURSOR 20, 7:PRINT"■■■---DATA ﾌﾞﾝ ｻｸｾｲ
2100 CURSOR 20, 8:PRINT"■-----DATA file ｻｸｾｲ
2110 CURSOR 20, 9:PRINT"■-----END
2200 REM*** DOT print ***
2210 FOR I= 0 TO C-1
2220 FOR J= 0 TO C-1
2240 CURSOR I,J:PRINT".";
2250 NEXT J,I
2260 RETURN
2500 REM*** CURSOR move ***
2505 GRAPHI1,O1:XX=X:YY=Y
2510 POSITION X*8,Y*8:PATTERN -8,C1$
2520 IF D=52 THEN X=X+(X>0):REM      ■
2530 IF D=54 THEN X=X-(X<(C-1)):REM ■
2540 IF D=56 THEN Y=Y+(Y>0):REM      ■
2550 IF D=50 THEN Y=Y-(Y<(C-1)):REM ■
2560 POSITION XX*8,YY*8:PATTERN -8,CO$
2570 POSITION X*8,Y*8:PATTERN -8,C1$
```

```
2580 RETURN
2800 REM***graphic pattern print ***
2810 GOSUB 4800:REM DOT DATA → PATTERN DATA
2820 POSITION 30*8,0:PATTERN -16,CO$+CO$
2830 POSITION 30*8,0:PATTERN -C,PAT$
2840 RETURN
3000 REM*** ALL CLEAR ***
3010 CURSOR 0,20:PRINT"ALL CLEAR ｼﾃ ﾖﾛｼｲ ﾃﾞｽｶ? (Yes or No)
3020 GET B$:IF B$="" THEN 3020
3030 IF B$="N" THENCURSOR 0,20:PRINT SPACE$(39);:GOTO 1030
3040 IF B$<>"Y" THEN 3020
3050 X=0:XX=0:Y=0:YY=0
3060 GOTO 1000
3300 REM*** DOT RESET ***
3310 CURSOR X,Y:PRINT".";
3320 RETURN
3500 REM*** DOT SET ***
3510 CURSOR X,Y:PRINT"●";
3520 RETURN
4000 REM*** DATA ｻｸｾｲ ***
4050 GOSUB 4800 :REM DOT → PATTERN
4060 IF N>20 THEN CURSOR 0,22:PRINT"ｺﾚｲｼﾞｮｳ ﾃﾞｷﾏｾﾝ":RETURN
4070 A$(N)=STR$(STAT)+"DATA"
4080 FLG=0
4090 FOR I=0 TO C-1
4100 IF FLG=0 THEN FLG=1:GOTO 4120
4110 A$(N)=A$(N)+","
4120 A$(N)=A$(N)+STR$(ASC(MID$(PAT$,I+1,1)))
4130 NEXT I
4140 CURSOR 0,18:PRINT A$(N):PRINT"N=";N
4150 N=N+1:STAT=STAT+10
4160 IF N>20 THEN PRINT"ｺﾚｲｼﾞｮｳ ﾃﾞｷﾏｾﾝ":RETURN
4170 RETURN
4400 REM---- DATA file ｻｸｾｲ ----
4410 ON ERROR GOTO 4900
4420 PRINT CHR$(6);:GRAPH O0
4430 INPUT "file name ? ";FILE$
4440 IF LEN(FILE$)>16 THEN 4430
4445 IF N=0 THEN PRINT"NO-DATA":GOSUB 2000:RETURN
4450 WOPEN #1,FD2@1,FILE$
4460 FOR I=0 TO N
4470 PRINT#1,A$(I)
4480 NEXT I
4490 CLOSE #1:N=0
4500 GOSUB 2000:REM ｶﾞﾒﾝ print
4510 RETURN
4800 REM  DOT → PATTERN
4805 PAT$="":PAT=0:K=0
4810 IF C=16 THEN FOR K=0 TO 8 STEP 8
4820 FOR I=0 TO C+K-1:PAT=0
4830 FOR J=0 TO 7
4840 IF CHARACTER$(J+K,I)="●" THEN PAT=PAT+2^(7-J)
4850 NEXT J
4860 PAT$=PAT$+CHR$(PAT)
4870 NEXT I
4880 IF C=16 THEN NEXT K
4890 RETURN
4900 REM---- ERROR ｼｮﾘ ----
4910 PRINT"ERROR ";ERN;" IN ";ERL
4920 USR($0F14):USR($0F14)
4930 PRINT" OK ?  hit any key "
4940 GET Z$:IF Z$="" THEN 4940
4950 RESUME
5000 REM*** program ｻｸｾｲ ***
5005 ON ERROR GOTO 4900
5010 CONSOLEC80,S0,24,N:GRAPH O0
5020 PRINTCHR$(6);
5030 INPUT "file name ? ";FILE$
5040 IF LEN(FILE$)>16 THEN 4430
5050 ROPEN #1,FD2@1,FILE$
5060 INPUT#1,A$
5070 PRINT A$
5080 IF EOF(#1) THEN 5100
5090 GOTO 5060
5100 CLOSE #1
5110 END
```

テープBASICの方は4450～4490，5000～5100を直してください。ここに示すのは一例です。
5005は削除してください。

```
4450 WOPEN/T FILE$
4470 PRINT/T A$(I)
4490 PRINT/T "END OF FILE":CLOSE/T:N=0
5005 ―――――― トル
5050 ROPEN/T FILE$
5060 INPUT/T A$
5080 IF A$="END OF FILE"THEN 5100
5100 CLOSE/T
```

MZ-700 (Hu-BASIC)
カセット・オペレーティング・プログラム
——CMT/OS VER.1.1——

大森 雅之

はじめに

MZ-700ファンのみなさん，こんにちは。Oh!MZ 3月号のIPL27（カセットテープ転送速度を1200ボーから2700ボーに）を利用していますか？ 私はBASIC のシステムテープ，マシン語テープのBACK UP など，便利に使っています。そこで今回，IPL27をさらに改造して，2700ボーテープの作成以外にも使えるようにしてみました。以下ご紹介したいと思います。

IPL27からの改造ポイント

1．IPL27を使ってSAVEされた2700ボーのテープもコピーできる。
2．作成するテープ速度を1200ボー，2700ボーと選択できる。
3．LOADINGするテープのINFORMATION BLOCKの内容を画面表示する。
4．SAVEする前にモニタに戻れるので，プログラムの内容変更ができる。
5．SAVE回数を1回にしたため，SAVE時間が短かい。また，複数プログラムの連続SAVE時に都合が良い。
6．B000～BFFF のバッファーエリアを必要としない。

上記1,2の機能により，1200ボー←→2700ボーテープの変換が自由にできます。また，3，4の機能により，内容不明なテープの解析やプログラムの変更が可能になります。

CMT/OSのコマンド説明

MZ-700のCMTについての説明および今回のプログラムの原理などについては，IPL27と同じです。本誌3月号を参考にしてください。ここでは，CMT/OSの仕様について説明します。

コマンド

[1]：1200ボー―INFORM READ
1200ボーが書かれたテープのINFORMATION BLOCKの内容を読み，画面に表示します。

[2]：1200ボー 1カイ SAVE
LOADINGしたプログラムを1200ボーで1回だけSAVEします。

[3]：2700ボー―INFORM READ
2700ボーで書かれたテープ（IPL27で作ったテープなど）のINFORMATION BLOCKをLOAD，表示します。

[4]：2700ボー 1カイ SAVE
LOADINGしたプログラムを2700ボーで1回だけSAVEします。

[5]：IPL27／X1 AND SAVE
IPLプログラムを1200ボーでSAVEしたあと，LOADINGしたプログラムを2700ボーでSAVEします。

[D]：DATA BLOCK READ
プログラムのDATA BLOCKを読み込みます。速度は[1]か[3]のコマンドでLOADINGしたものと同じになります。

[R]：RETURN ROM MONITOR
MONITOR 1Z-009A(ROM)に戻ります。

[M]：RETURN MONITOR 27
2700ボー―MONITOR(RAM)に戻ります。

CMT/OSの使い方

ダンプリスト（リスト2）にしたがって，プログラムを入力してください。アセンブラを持っている方は，ソースレベルで入力したほうが，アドレスの変更，プログラムの拡張など，後のち便利です。直接マシン語で作る方は，MONITOR 1Z-009A のMコマンドでプログラム（$A000～$A2FF）を入力し，Sコマンドで SAVEしてください。プログラムのスタートアドレスは$A000です。

〈使用例1〉マシン語プログラムのBACK UP作成

1．MONITOR 1Z-009Aより，CMT/OSプログラムをLOAD，スタートする。
2．コピーするプログラムのボーレートに合わせて，[1]または[3]キーを押す。
3．テープをセットし（OKなら[Y]キー），PLAYボタンを押す（INFORMATION BLOCKのLOADING）。
4．READYになったら[D]キーを押す（DATA BLOCKのLOADING）。
5．READYになったら新しいカセットをセットし（OKなら[Y]キーを押す），SAVEしたいボーレートに合わせて，[2][3][4]のいずれかのキーを押す。
6．READYになったら，[R]または[M]コマンドでモニタに戻る。

〈使用例2〉Hu-BASICを2700ボーに改造する

1．CMT/OS起動。
2．[1]コマンドでHu-BASIC（マスターテープ）のINFORMATION BLOCKを読む。
3．[D]コマンドでHu-BASICのDATA BLOCKを読む。
4．[R]コマンドでモニタに戻る。
5．モニタの[M]コマンドで次のメモリ内容を変更する。

ADDRESS	DATE	
$2021	$6F ⟶ $27	ボーレート変更
$202E	$6F ⟶ $27	
$2040	$36 ⟶ $13	
$204A	$34 ⟶ $12	
$2178	$4F ⟶ $1D	
$1FC0	$02 ⟶ $01	SAVE回数変更
$20A1	$02 ⟶ $01	
$2116	$02 ⟶ $01	
$A30B	$0D ⟶ $20	プログラム名変更（Hu-BASIC2701）
$A30C	$00 ⟶ $32	
$A30D	$00 ⟶ $37	
$A30E	$00 ⟶ $30	
$A30F	$00 ⟶ $31	
$A310	$00 ⟶ $0D	

6．モニタ[J]コマンドで CMT/OS に戻る (JA000)。
7．CMT/OS [5]コマンドでSAVEする。
8．終了

以上で，2700ボー，1回SAVEのHu-BASICテープができあがります。

注）1回SAVEなので，プログラムのVERIFYおよびBACK UPテープの作成を忘れずに！ 2回SEVEしたい人は，$1FC0，$20A1，$2116のメモリ変更をしなければ，2回SAVEとなります。

また，カセットテープはできるだけ良質のものを使用するようにしたほうが良いでしょう。マイコン用テープならOKです。

なお，2700ボーHu-BASIC上で，1200ボーのプログラムをLOADするときは，直接実行文で次の操作をすれば可能です。

• • •

(2700ボー⟶1200ボー変更時)

POKE 3617, 111 : POKE 3630, 111 : POKE 3648, 54:POKE 3658, 52:POKE 3960, 79 [CR]

(1200ボーから2700ボーに戻るには)

POKE 3617, 39:POKE 3630, 39:POKE 3648, 19:POKE 3658, 18:POKE 3960, 29 [CR]

また，2700ボーHu-BASICテープのあとに，Hu-BASICサブルーチン(リスト3)をSAVEしておくと，1200ボー⟷2700ボーが自由に変更できるので便利です。

○操作方法……直接実行文で以下の命令を実行する。

1200ボーにするには　CALL &HF000
2700ボーにするには　CALL &HF002

リスト1

```
                   ;******************************
                   ;* MZ-700  CMT/OS   VER(1.1) *
                   ;* (IPL LOADER 27/X1)        *
                   ;*                            *
                   ;*   '83/6/26   BY   M.OMORI *
                   ;******************************
                   ;----- SYSTEM CONSTANT &LINK -----
0000               MONIT   EQU    0000H   ;MONIT ON
0009               NL      EQU    0009H   ;NEW LINE
0012               PRNT    EQU    0012H   ;1CH PRINT
0015               MSG     EQU    0015H   ;1 LINE PR
001B               GETKY   EQU    001BH   ;GET KEY
0021               WRINF   EQU    0021H   ;WRITE INF
0024               WRDAT   EQU    0024H   ;WRITE DA.
0027               RDINF   EQU    0027H   ;READ INF.
002A               RDDAT   EQU    002AH   ;READ DATA
003E               BELL    EQU    003EH   ;BELL ON
00AD               ST1     EQU    00ADH   ,MONITOR
00FE               ?ER     EQU    00FEH   ;ERROR(L)
0108               LOAD    EQU    0108H   ;L COMMAND
03B4               PR[HL]  EQU    03B4H   ;(HL)DISP
05FA               NLPHL   EQU    05FAH   ;NL+PR HL
0920               ?PRTS   EQU    0920H   ;SP.PRINT
09B3               ??KEY   EQU    09B3H   ;KEY IN
0BCE               ?DACN   EQU    0BCEH   ;DP ASCII
0DDC               ?DPCT   EQU    0DDCH   ;CRTCTRL
013E               MSG?2   EQU    013EH   ;'LOADING'
0942               MSGOK   EQU    0942H   ;'OK!'
10F0               ATRB    EQU    10F0H   ;ATTRIBUTE
10F1               NAME    EQU    10F1H   ;FILE NAME
1102               SIZE    EQU    1102H   ;BYTE SIZE
1104               DTADR   EQU    1104H   ;DATA ADR
1106               EXADR   EQU    1106H   ;EX. ADR
1108               COMNT   EQU    1108H   ;COMMENT
                   ;**********************
                   ;*    MAIN PROGRAM    *
                   ;**********************
0000                       ORG    0A000H
A000 CD0CA1                CALL   TITLE   ;TITLE PR
A003 CD93A0        LOOP:   CALL   RDYMSG  ;READY PR
A006 CDB309        KEY:    CALL   ??KEY   ; KEY IN
A009 CDCE0B                CALL   ?DACN
A00C CD1200                CALL   PRNT    ;KET PR
A00F FE31                  CP     '1'     ;-
A011 2823                  JR     Z,RD12  ;-1200INF.
A013 FE32                  CP     '2'     ;*
A015 2829                  JR     Z,SV12  ;*1200SAV.
A017 FE33                  CP     '3'     ;-
A019 2820                  JR     Z,RD27  ;-2700INF.
A01B FE34                  CP     '4'     ;*
A01D 283F                  JR     Z,SV27  ;*2700SAV.
A01F FE35                  CP     '5'     ;-
A021 2825                  JR     Z,SVIPL ;-IPL+SAVE
A023 FE44                  CP     'D'     ;*
A025 2857                  JR     Z,READDT;*READ DT
A027 FE52                  CP     'R'     ;-
A029 285E                  JR     Z,RETMON;-ROM MON
A02B FE4D                  CP     'M'     ;*
A02D 285E                  JR     Z,RETRAM;*MON2701
A02F 3EC7                  LD     A,0C7H
A031 CDDC0D                CALL   ?DPCT   ;DEL
A034 18D0                  JR     KEY     ;-
                   ;------- INFORMATION READ --------
A036 CD34A2        RD12:   CALL   READ12  ;1200ﾎﾞ-
A039 18C8                  JR     LOOP    ;RDINF
A03B CD2FA2        RD27:   CALL   READ27  ;2700ﾎﾞ-
A03E 18C3                  JR     LOOP    ;RDINF
                   ;-------- PROGRAM  SAVE ----------
A040 CDA9A0        SV12:   CALL   TPMSG   ;
A043 CDF9A0        SV12:   CALL   M1201   ;1200ﾎﾞ-
A046 181C                  JR     COPY    ;
A048 CDA9A0        SVIPL:  CALL   TPMSG
A04B CDF9A0                CALL   M1201
A04E 11F010                LD     DE,ATRB ;IPL 27X1
A051 21E8A2                LD     HL,DATA1;DATA
A054 011800                LD     BC,0018H;
A057 EDB0                  LDIR           ;ATRB
A059 CDBBA0                CALL   SAV     ;IPL SAVE
A05C 1803                  JR     CP27
A05E CDA9A0        SV27:   CALL   TPMSG
A061 CDCEA0        CP27:   CALL   M2701   ;2700ﾎﾞ-
A064 11F010        COPY:   LD     DE,ATRB ;ATRB
A067 2100A3                LD     HL,BUF1 ; SET
A06A 012000                LD     BC,0020H;
A06D EDB0                  LDIR           ;
A06F CDBBA0                CALL   SAV     ;PRG SAVE
A072 CD0900                CALL   NL
A075 114209                LD     DE,MSGOK
A078 DF                    RST    18H
A079 CD3E00                CALL   BELL
A07C 1885          LPB:    JR     LOOP
                   ;---------- READ DATA -----------
A07E CDA9A0        READDT: CALL   TPMSG
A081 CD2A00                CALL   RDDAT   ;DATA RD
A084 DAFE00                JP     C,?ER
A087 18F3                  JR     LPB
                   ;------- RETURN TO MONITOR -------
```

```
A089 D3E2      RETMON: OUT    (0E2H),A;ROM CH
A08B 1803              JR     MON     ;JP MONIT.
A08D CDCEA0    RETRAM: CALL   M2701   ;M27 CH
A090 C3AD00    MON:    JP     ST1 ;JUMP MONITOR

               ;--------------------------------
               ;******************
               ;*  SUB ROUTINE  *
               ;******************

               ;--------- READY MSG -----------

A093 CD0900    RDYMSG: CALL   NL
A096 11A3A0            LD     DE,RMSG
A099 CD1500            CALL   MSG
A09C CD0900            CALL   NL
A09F CD3E00            CALL   BELL
A0A2 C9                RET

A0A3 52454144  RMSG:   DEFM   'READY'
A0A7 59
A0A8 0D                DEFB   0DH

               ;------ TAPE SET MSG ------------

A0A9 CD0900    TPMSG:  CALL   NL
A0AC 11D4A2            LD     DE,MSG2
A0AF DF                RST    18H
A0B0 CD3E00            CALL   BELL
A0B3 CD1B00    L03:    CALL   GETKY   ;Y KEY WA.
A0B6 FE59              CP     59H
A0B8 20F9              JR     NZ,L03
A0BA C9                RET

               ;------ SAVE INF & DATA ----------

A0BB CD2100    SAV:    CALL   WRINF   ;WRINF
A0BE DAFE00            JP     C,?ER
A0C1 CD2400            CALL   WRDAT   ;WRDATA
A0C4 DAFE00            JP     C,?ER
A0C7 C9                RET

               ;--------------------------------

               ;  ******** IPL 27X1 ********

A0C8 CDCEA0    ST:     CALL   M2701   ;
A0CB C30801            JP     LOAD    ;L COM.

               ;  ------ ROM   RAM SUB ------

A0CE 210000    M2701:  LD     HL,0000H ;ADD.SET
A0D1 010010            LD     BC,1000H ;
A0D4 D3E2      TR1:    OUT    (0E2H),A ;
A0D6 7E                LD     A,(HL)   ;ROM ACC
A0D7 D3E0              OUT    (0E0H),A ;
A0D9 77                LD     (HL),A   ;ACC RAM
A0DA EDA1              CPI             ;
A0DC EAD4A0            JP     PE,TR1   ;NEXT

A0DF 3E01              LD     A,01     ;
A0E1 328E04            LD     (048EH),A ;1ｶｲSAVE
A0E4 321205            LD     (0512H),A ;1ｶｲREAD
A0E7 32B105            LD     (05B1H),A ;1ｶｲVERI
A0EA 3E0B      SAVE27: LD     A,0BH    ;
A0EC 325A07            LD     (075AH),A ;2700ﾎﾞ-
A0EF 3D                DEC    A        ;ﾆ ﾍﾝｺｳ
A0F0 326107            LD     (0761H),A ;
A0F3 3E22              LD     A,22H    ;
A0F5 324B0A            LD     (0A4BH),A ;
A0F8 C9                RET

               ;------ 1200ﾎﾞ- RAM MONITOR-------

A0F9 CDCEA0    M1201:  CALL   M2701    ;M27
A0FC 3E19      SAVE12: LD     A,19H    ;
A0FE 325A07            LD     (075AH),A ;1200ﾎﾞ-
A101 3E17              LD     A,17H    ;ﾆ ﾍﾝｺｳ
A103 326107            LD     (0761H),A ;
A106 3E53              LD     A,53H    ;
A108 324B0A            LD     (0A4BH),A ;
A10B C9                RET

               ;-------- TITLE PRINT SUB --------

A10C 1116A1    TITLE:  LD     DE,TITLDT  ;CLS
A10F CD1500            CALL   MSG        ;ﾀｲﾄﾙPR
A112 CD0900            CALL   NL
A115 C9                RET

A116 1611      TITLDT: DEFB 22,17        ;CLS
A118 20434D54          DEFM ' CMT/OS   VER(1.0)'
A11C 2F4F5320
A120 20205645
A124 5228312E
A128 3029
A12A 13131311          DEFB   19,19,19,17,75,69,89 ;KEY
A12E 4B4559
A131 15111111          DEFB   21,17,17,17
A135 20313230          DEFM ' 1200ﾎﾞ-INFORM READ--(1)'
A139 30AEBE90
A13D 494E464F
A141 524D2052
A145 4541442D
A149 2D283129
A14D 15111111          DEFB   21,17,17,17,17
A151 11
A152 20313230          DEFM ' 1200ﾎﾞ-  1ｶｲ  SAVE--(2)'
A156 30AEBE90
A15A 20203196
A15E 92202053
A162 4156452D
A166 2D283229
A16A 15111111          DEFB   21,17,17,17,17,17
A16E 1111
A170 20323730          DEFM ' 2700ﾎﾞ-INFORM READ--(3)'
A174 30AEBE90
A178 494E464F
A17C 524D2052
A180 4541442D
A184 2D283329
A188 15111111          DEFB   21,17,17,17,17,17,17
A18C 111111
A18F 20323730          DEFM ' 2700ﾎﾞ-  1ｶｲ  SAVE--(4)'
A193 30AEBE90
A197 20203196
A19B 92202053
A19F 4156452D
A1A3 2D283429
A1A7 15111111          DEFB   21,17,17,17,17,17,17,17
A1AB 11111111
A1AF 2049504C          DEFM ' IPL 27/X1 AND SAVE--(5)'
A1B3 2032372F
A1B7 58312041
A1BB 4E442053
A1BF 4156452D
A1C3 2D283529
A1C7 15111111          DEFB   21,17,17,17,17,17,17,17,17
A1CB 11111111
A1CF 11
A1D0 20444154          DEFM ' DATA BLOCK READ ----(D)'
A1D4 4120424C
A1D8 4F434B20
A1DC 52454144
A1E0 202D2D2D
A1E4 2D284429
A1E8 15111111          DEFB   21,17,17,17,17,17,17,17,17
A1EC 11111111
A1F0 11
A1F1 11                DEFB   17
A1F2 20524554          DEFM ' RETURN ROM MONITOR--(R)'
A1F6 55524E20
A1FA 524F4D20
A1FE 4D4F4E49
A202 544F522D
A206 2D285229
A20A 15111111          DEFB   21,17,17,17,17,17,17,17,17
A20E 11111111
A212 11
A213 1111              DEFB   17,17
A215 20524554          DEFM ' RETURN MONITOR 27 --(M)'
```

```
     A219 55524E20
     A21D 4D4F4E49
     A221 544F5220
     A225 3237202D
     A229 2D284D29
     A22D 110D          DEFB   17,0DH

                     ;------- INFORM READ+PRINT -------

     A22F CDCEA0   READ27: CALL   M2701     ;MON27
     A232 1802             JR     IFREAD
     A234 D3E2     READ12: OUT    (0E2H),A  ;ROM CH
     A236 CDA9A0   IFREAD: CALL   TPMSG
     A239 CD2700           CALL   RDINF     ;INF RD
     A23C DAFE00           JP     C,?ER
     A23F CD0CA1           CALL   TITLE     ;CLS
     A242 118FA2           LD     DE,MSG11  ;- NANE
     A245 DF               RST    18H       ;-
     A246 11F110           LD     DE,NAME   ;-
     A249 DF               RST    18H       ;-
     A24A CD0900           CALL   NL        ;-
     A24D 119CA2           LD     DE,MSG12  ;* ARRB
     A250 DF               RST    18H       ;*
     A251 21F010           LD     HL,ATRB   ;*
     A254 CD88A2           CALL   PRDATA1   ;*
     A257 11AAA2           LD     DE,MSG13  ;- SIZE
     A25A DF               RST    18H       ;-
     A25B 210311           LD     HL,SIZE+1 ;-
     A25E CD84A2           CALL   PRDATA2   ;-
     A261 11B8A2           LD     DE,MSG14  ;* DTADR
     A264 DF               RST    18H       ;*
     A265 210511           LD     HL,DTADR+1;*
     A268 CD84A2           CALL   PRDATA2   ;*
     A26B 11C6A2           LD     DE,MSG15  ;- EXADR
     A26E DF               RST    18H       ;-
     A26F 210711           LD     HL,EXADR+1;-
     A272 CD84A2           CALL   PRDATA2   ;-
     A275 CD0900           CALL   NL
     A278 1100A3           LD     DE,BUF1
     A27B 21F010           LD     HL,ATRB
     A27E 012000           LD     BC,0020H
     A281 EDB0             LDIR
     A283 C9               RET

     A284 CDB403   PRDATA2:CALL   PR[HL]    ;INFORM
     A287 2B               DEC    HL        ;DATA
     A288 CDB403   PRDATA1:CALL   PR[HL]    ;PRINT
     A28B CD0900           CALL   NL        ;
     A28E C9               RET

     A28F 2046494C MSG11:  DEFM    ' FILE NAME :'
     A293 45204E41
     A297 4D45203A
     A29B 0D               DEFB    0DH
     A29C 20415454 MSG12:  DEFM    ' ATTRIBUTE :$'
     A2A0 52494255
     A2A4 5445203A
     A2A8 24
     A2A9 0D               DEFB    0DH
     A2AA 20424954 MSG13:  DEFM    ' BITE SIZE :$'
     A2AE 45205349
     A2B2 5A45203A
     A2B6 24
     A2B7 0D               DEFB    0DH
     A2B8 20444154 MSG14:  DEFM    ' DATA ADR. :$'
     A2BC 41204144
     A2C0 522E203A
     A2C4 24
     A2C5 0D               DEFB    0DH
     A2C6 20455820 MSG15:  DEFM    ' EX ADDRESS:$'
     A2CA 41444452
     A2CE 4553533A
     A2D2 24
     A2D3 0D               DEFB    0DH

                   ;--------------------------------

     A2D4 20544150 MSG2:   DEFM ' TAPE SET OK ?(Y/N)'
     A2D8 45205345
     A2DC 54204F4B
     A2E0 203F2859
     A2E4 2F4E29
     A2E7 0D               DEFB    0DH

     A2E8 01       DATA1:  DEFB    01H
     A2E9 49504C20         DEFM   'IPL LOADER 27/X1'
     A2ED 4C4F4144
     A2F1 45522032
     A2F5 372F5831
     A2F9 0D               DEFB    0DH
     A2FA 3100             DEFW    M1201-ST
     A2FC C8A0             DEFW    ST
     A2FE C8A0             DEFW    ST

     A300          BUF1:   DEFS    20H

     A320                  END

ERRORS    0
```

リスト2

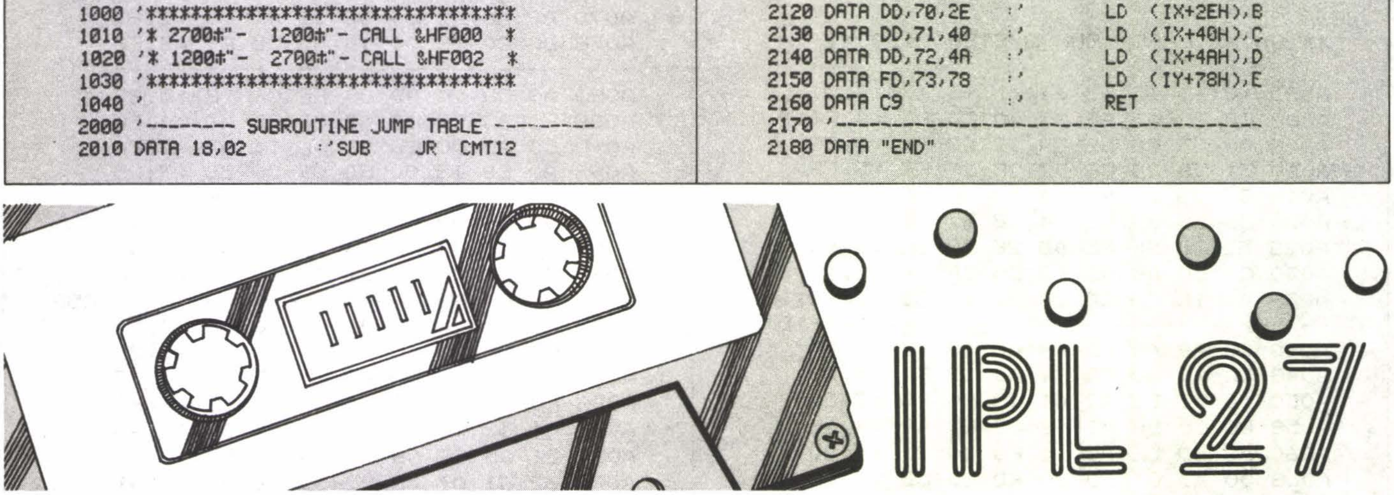
```
** OH!MZ CHECK SUM SYSTEM V1.0 **

ADR. +0 +1 +2 +3 +4 +5 +6 +7 :SUM.
A000 CD 0C A1 CD 93 A0 CD B3 :4.FA
A008 09 CD CE 0B CD 12 00 FE :3.8C
A010 31 28 23 FE 32 28 29 FE :2.FB
A018 33 28 20 FE 34 28 3F FE :3.12
A020 35 28 25 FE 44 28 57 FE :3.41
A028 52 28 5E FE 4D 28 5E 3E :2.E7
A030 C7 CD DC 0D 18 D0 CD 34 :4.66
A038 A2 18 C8 CD 2F A2 18 C3 :3.FB
------------------------------------1D1C
A040 CD A9 A0 CD F9 A0 18 1C :4.B0
A048 CD A9 A0 CD F9 A0 11 F0 :5.7D
A050 10 21 E8 A2 01 18 00 ED :2.C1
A058 B0 CD BB A0 18 03 CD A9 :4.69
A060 A0 CD CE A0 11 F0 10 21 :4.0D
A068 00 A3 01 20 00 ED B0 CD :3.2E
A070 BB A0 CD 09 00 11 42 09 :2.8D
A078 DF CD 3E 00 18 85 CD A9 :3.FD
------------------------------------1F1C
A080 A0 CD 2A 00 DA FE 00 18 :3.87
A088 F3 D3 E2 18 03 CD CE A0 :4.FE
A090 C3 AD 00 CD 09 00 11 A3 :2.FA
A098 A0 CD 15 00 CD 09 00 CD :3.25
A0A0 3E 00 C9 52 45 41 44 59 :2.7C
A0A8 0D CD 09 00 11 D4 A2 DF :3.49
A0B0 CD 3E 00 CD 1B 00 FE 59 :3.4A
A0B8 20 F9 C9 CD 21 00 DA FE :4.A8
------------------------------------1C5B
A0C0 00 CD 24 00 DA FE 00 C9 :3.92
A0C8 CD CE A0 C3 08 01 21 00 :3.28
A0D0 00 01 00 10 D3 E2 7E D3 :3.17
A0D8 E0 77 ED A1 EA D4 A0 3E :5.81
A0E0 01 32 8E 04 32 12 05 32 :1.40
A0E8 B1 05 3E 0B 32 5A 07 3D :1.CF
A0F0 32 61 07 3E 22 32 4B 0A :1.81
```

```
A0F8 C9 CD CE A0 3E 19 32 5A :3.E7
-----------------------------------17C9
A100 07 3E 17 32 61 07 3E 53 :1.87
A108 32 4B 0A C9 11 16 A1 CD :2.E5
A110 15 00 CD 09 00 C9 16 11 :1.DB
A118 20 43 4D 54 2F 4F 53 20 :1.F5
A120 20 20 56 45 52 28 31 2E :1.B4
A128 30 29 13 13 13 11 4B 45 :1.33
A130 59 15 11 11 11 20 31 32 :1.24
A138 30 30 AE BE 90 49 4E 46 :3.39
-----------------------------------0F80
A140 4F 52 4D 20 52 45 41 44 :2.2A
A148 2D 2D 28 31 29 15 11 11 :1.13
A150 11 11 20 31 32 30 30 AE :1.B3
A158 BE 90 20 20 31 96 92 20 :3.07
A160 20 53 41 56 45 2D 2D 28 :1.D1
A168 32 29 15 11 11 11 11 11 :0.C5
A170 20 32 37 30 30 AE BE 90 :2.E5
A178 49 4E 46 4F 52 4D 20 52 :2.3D
-----------------------------------0FAF
A180 45 41 44 2D 2D 28 33 29 :1.A8
A188 15 11 11 11 11 11 11 20 :0.9B
A190 32 37 30 30 AE BE 90 20 :2.E5
A198 20 31 96 92 20 20 53 41 :2.4D
A1A0 56 45 2D 2D 28 34 29 15 :1.8F
A1A8 11 11 11 11 11 11 11 20 :0.97
A1B0 49 50 4C 20 32 37 2F 58 :1.F5
A1B8 31 20 41 4E 44 20 53 41 :1.D8
-----------------------------------0D68
A1C0 56 45 2D 2D 28 35 29 15 :1.90
A1C8 11 11 11 11 11 11 11 11 :0.88
A1D0 20 44 41 54 41 20 42 4C :1.E8
A1D8 4F 43 4B 20 52 45 41 44 :2.19
A1E0 20 2D 2D 2D 2D 28 44 29 :1.69
A1E8 15 11 11 11 11 11 11 11 :0.8C
A1F0 11 11 20 52 45 54 55 52 :1.D4
A1F8 4E 20 52 4F 4D 20 4D 4F :2.18
-----------------------------------0BFA
A200 4E 49 54 4F 52 2D 2D 28 :2.0E
A208 52 29 15 11 11 11 11 11 :0.E5
A210 11 11 11 11 11 20 52 45 :1.0C
A218 54 55 52 4E 20 4D 4F 4E :2.53
A220 49 54 4F 52 20 32 37 20 :1.E7
A228 2D 2D 28 4D 29 11 0D CD :1.E3
A230 CE A0 18 02 D3 E2 CD A9 :4.B3
A238 A0 CD 27 00 DA FE 00 CD :4.39
-----------------------------------1308
A240 0C A1 11 8F A2 DF 11 F1 :3.D0
A248 10 DF CD 09 00 11 9C A2 :3.14
A250 DF 21 F0 10 CD 88 A2 11 :4.08
A258 AA A2 DF 21 03 11 CD 84 :3.B1
A260 A2 11 B8 A2 DF 21 05 11 :3.23
A268 CD 84 A2 11 C6 A2 DF 21 :4.6C
A270 07 11 CD 84 A2 CD 09 00 :2.E1
A278 11 00 A3 21 F0 10 01 20 :1.F6
-----------------------------------1B03
A280 00 ED B0 C9 CD B4 03 2B :4.15
A288 CD B4 03 CD 09 00 C9 20 :3.43
A290 46 49 4C 45 20 4E 41 4D :2.1C
A298 45 20 3A 0D 20 41 54 54 :1.B5
A2A0 52 49 42 55 54 45 20 3A :2.25
A2A8 24 0D 20 42 49 54 45 20 :1.95
A2B0 53 49 5A 45 20 3A 24 0D :1.C6
A2B8 20 44 41 54 41 20 41 44 :1.DF
-----------------------------------1288
A2C0 52 2E 20 3A 24 0D 20 45 :1.70
A2C8 58 20 41 44 44 52 45 53 :2.2B
A2D0 53 3A 24 0D 20 54 41 50 :1.C3
A2D8 45 20 53 45 54 20 4F 4B :2.0B
A2E0 20 3F 28 59 2F 4E 29 0D :1.93
A2E8 01 49 50 4C 20 4C 4F 41 :1.E2
A2F0 44 45 52 20 32 37 2F 58 :1.EB
A2F8 31 0D 31 00 C8 A0 C8 A0 :3.3F
-----------------------------------1008
```

リスト3

```
100 REM ***********************
110 REM * HU BASIC SUBROUTINE *
120 REM *                     *
130 REM * '83/7/10    M.OMORI *
140 REM ***********************
150 REM
160 LIMIT &HEFFF:SA=&HF000
170 READ SD$:IF SD$="END" THEN END
180 S1=VAL("&H"+SD$):POKE@ SA,S1:SA=SA+1:GOTO 170
190 '---------------------------------------
1000 '*********************************
1010 '* 2700ﾎﾞｰ  1200ﾎﾞｰ CALL &HF000  *
1020 '* 1200ﾎﾞｰ  2700ﾎﾞｰ CALL &HF002  *
1030 '*********************************
1040 '
2000 '-------- SUBROUTINE JUMP TABLE ---------
2010 DATA 18,02       :'SUB    JR  CMT12
2020 DATA 18,08       :'       JR  CMT27
2030 '----------- CMT12,CMT27 -----------
2040 DATA 01,36,6F    :'CMT12  LD  BC,6F36H
2050 DATA 11,4F,34    :'       LD  DE,344FH
2060 DATA 18,06       :'       JR  CMTCH
2070 DATA 01,13,27    :'CMT27  LD  BC,2713H
2080 DATA 11,1D,12    :'       LD  DE,121DH
2090 DATA DD,21,00,0E:'CMYCH  LD  IX,0E00H
2100 DATA FD,21,00,0F:'       LD  IY,0F00H
2110 DATA DD,70,21    :'       LD  (IX+21H),B
2120 DATA DD,70,2E    :'       LD  (IX+2EH),B
2130 DATA DD,71,40    :'       LD  (IX+40H),C
2140 DATA DD,72,4A    :'       LD  (IX+4AH),D
2150 DATA FD,73,78    :'       LD  (IY+78H),E
2160 DATA C9          :'       RET
2170 '-----------------------------------
2180 DATA "END"
```



MZ-731 (Hu-BASIC)
漢字作成プログラム

佐藤 裕二

MZ-731に付属のX-Y プロッタプリンタの特長を生かして，漢字の作成および処理用プログラムを作ってみました。私を含め，MZ-731をスモールビジネスなどに使おうと思っている方には，便利だと思います。

漢字の作成

①作図

リスト1のプログラムをRUNすると，何をするかを聞いてきます。メニューにより，1を入力すると，入れたい漢字の名前とデータ文の行番号を聞いてきます。このとき，行番号は本プログラムの最後の行より大きい数を入れてください。

入力が終わると，画面に正方形の方眼が表示されます。カーソルキーを任意の位置に移動させ，表1のキーを使って，書きたい漢字を書いてください。

作図が終わったら X キーを押します。

表1

KEY	PRINT
−	−
;	|
P	／
O	＼
+	+
ｽﾍﾟｰｽｷｰ	ｼｭｳｾｲ

②座標のインプット

カーソルを一画の初め，変化点，および終わりに移動させ，それぞれに対して Z キーを押します。Zを押した箇所には＊が入ります。

ひとつの画の座標をインプットし終えたら A キーを押します。この作業を，すべての画に対して行なってください。もし，途中でキーを押しまちがえたときは，カーソルを上方向に上げると，絵はそのままで，座標のインプットを初めからやり直すことができます。

すべての画のインプットが終わったら，E キーを押してください。

③データをプログラムに組み込む

E キーを押すと，画面下に行番号とデータ文の形式で，今打ち込んだ漢字のデータが表示されます。そこで，カーソルを行番号の頭にもっていき CR を押してください。これを忘れると，今作った漢字のデータがプログラム中に残らないことになってしまいます。

④プリンタへの印字

もう一度プログラムをRUNさせて，メニューの2を押します。するとプリンタに今入力した漢字が書かれます(図1)。もし，字の形が気に入らなかったら，もう一度RUNさせてからメニューの3を押します。そうすると，プリンタに5単位の方眼中に文字が印字されます(図2)。DATA文の内容は図3のようになっていますから，それによってデータの修正を行なってください。

なお，画数および画のデータ数は画面に表示されます。

気に入った字ができたら，データ文の先頭に' (リマーク) を付けておきます (リスト1の1000行参照)。

この作業を繰り返して，書きたい漢字のデータを作っていってください。

図1



図2

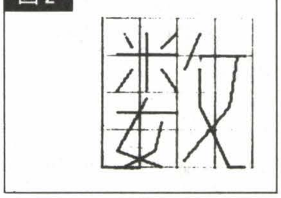


図3

```
                    画数
                     | 画のデータ数                      画のデータ数
                     |  ↓        X軸 Y軸                  ↓
5300 DATA ﾃﾝ,4,3,3,1,9,7,10,15,2,4,15,16
,15,2,15,10,5,10,3,9,7,13,3,17,1
    ↑                    ↑
```

漢字処理ルーチンの作成

書きたい漢字のデータを一応すべて入れ終わったら，各データ文に付けていたリマーク' をすべて削除します。次に，データ文だけを残して，他の行 (10−860行) はすべて消去してください (マニュアル P105 DELETE)。

RENUM 5000を実行して (マニュアルP108)，行番号を大きくしておいてから，アスキー形式でテープにSAVEしておきます (マニュアルP108)。

次に，リスト2の漢字サブルーチンを打ち込み，同じようにアスキー形式でテープにSAVEしてください。

漢字表示プログラムへの融合

①作業

漢字処理を行なおうとする本プログラムに，リスト2の漢字サブルーチンと，漢字のデータを付け加えます (マニュアルP107 MERGE)。

次に，漢字サブルーチンの3100行のRESTOREのあとに，データ文の行番号を入れていきます (リスト3の3100行)。これで，行番号を入れた順に1，2，3…と，漢字に番号が付くわけです (マニュアルP116 ON〜RESTORE)。

②説明

このプログラムの使用例として，リスト3を載せておきます。なお，ここで使われる変数は表2のとおりです。

このプログラムでは，漢字は LPRINT文と併用して使うことができますが，漢字

まじり文は1行を越えて書くことができません。そこで，1行を越える場合は，文を途中で区切って（RET＝0を入れる）新たに次の行から書き始めるようにしてください。なお，このプログラムは，TNモード時のみ有効です。

あとがき

漢字を使ったプログラムなどをたびたび作るという方は，漢字のデータを一度にたくさん作っておいてから，いるものだけをその都度取り出して使うようにすると使い良いのではないでしょうか。

以上，操作方法はややめんどうですが，フルセットでわずか130,000円弱のコンピュータで漢字が使えるようになるのですから，挑戦してみる価値は十分あると思うのですが，いかがでしょうか。

表2

変　数	説　　明	変　数	説　　明
KSTI=a	漢字の大きさを決める。 a＝ 0…普通文字　a＝ 1…拡大文字 a＝ −0.5…縮小文字 （拡大・縮小しない場合は省略してもよい）	RET＝0	a…順番　b…コード （b＝ 0の場合は空白を意味する） 漢字まじり文の改行を行なうとき，その都度入れる。
KAZ=a	漢字のひとつながりの数の指定。	a. bはパラメータ 注意事項）KAZを10以上に指定するときは，本プログラムの初めにKNの配列宣言を行なってください。	
KN(a)=b	漢字のひとつながりの中での順番と，その漢字のコードを指定。		

リスト1

```
10 '*****************************
20 '*******カンシ" サクセイ(CRT)******
30 '*****************************
40 ON ERROR GOTO690
50 MODE TN
60 '**メニュー**
70 COLOR7,1:CLS
80 CLS:PRINT"■  イマカラ カンシ"ノ サクセイ ヲ シマス。■■"
90 PRINT" カンシ"ヲ ツクルナラハ" =1":PRINT" フ゜リンター ニ カクナラ =2":PRINT" サ"ヒョウヲ ミタイナラハ"=3 "
100 INPUT GT
110 IF GT=1 THEN 120 ELSE IF GT=2 THEN 720 ELSE IF GT=3 THEN 810 ELSE 100
120 PRINTCHR$(19):INPUT"ナマエ  ";NA$
130 PRINTCHR$(21):INPUT"キ"ョウハ"ンコ"ウ";GYU
140 PRINT"■■  イマカラ ハシ"メマス。 ヨケレハ" [Y] ヲ オシテ クタ"サイ"
150 W$=INKEY$:IF W$="" THEN 150
160 '**ワク+セツメイ**
170 CLS:COLOR4,1
180 FOR I=1TO20
190 CURSORO,I:PRINT"▒":CURSOR20,I:PRINT"▒":CURSORI,0:PRINT"▒":CURSORI-1,21:PRINT
"▒"
200 NEXT:COLOR2
210 FOR I=5 TO 15 STEP 5
220 FOR N=1TO20:CURSORI,N:PRINT"!":NEXTN
230 FOR N=1TO19:CURSORN,I:PRINT"-":NEXT:CURSORO,O
240 NEXTI:COLOR7
250 CURSOR24,19:PRINT"サクス" カ" テ"キタラ X"
260 CURSOR24,21:PRINT"サ"ヒョウ ノ INPUT Z"
270 CURSOR24,22:PRINT"ヘン カ"テ"キタラ     A"
280 CURSOR24,23:PRINT"オワッタラ          E":CURSORO,O
290 '**サクス"**
300 B$=INKEY$(1)
310 IF B$="■" THEN Y=Y-1:CURSORX,Y
320 IF B$="■" THEN X=X+1:CURSORX,Y
330 IF B$="■" THEN Y=Y+1:CURSORX,Y
340 IF B$="■" THEN X=X-1:CURSORX,Y
350 IF B$="-" THEN CURSORX,Y:PRINT"-":GOTO300
360 IF B$=";" THEN CURSORX,Y:PRINT"|":GOTO300
370 IF B$="P" THEN CURSORX,Y:PRINT"/":GOTO300
380 IF B$="O" THEN CURSORX,Y:PRINT"\":GOTO300
390 IF B$="+" THEN CURSORX,Y:PRINT"+":GOTO300
400 IF B$=" " THEN CURSORX,Y:PRINT" ":GOTO300
410 IF B$<>"X" THEN 300
420 CURSOR22,0:PRINT"*サ"ヒョウ ヲ トル*":CURSORO,O:PRINT"▒"
430 '**サ"ヒョウ ヲ トル**
440 DIM W(200)
450 X=0:Y=0:U=2
460 FOR J=1 TO 30
470 IF J MOD7=0 THEN CONSOLEO,18,22,17:CLS:CONSOLE 0,25,0,40
480 IF J>=7 THEN JA=INT(J/7):JJ=J-7*JA ELSE JJ=J
490 U=U+1:IF J<>0 THEN CURSOR22,JJ*2:PRINT"ヘン ノ カス""
500 FOR K=1 TO 15
510 A$=INKEY$(1)
520 IF A$="■" THEN Y=Y-1:CURSORX,Y:GOTO510
530 IF A$="■" THEN X=X+1:CURSORX,Y:GOTO510
```

```
540 IF A$="■" THEN Y=Y+1:CURSORX,Y:GOTO510
550 IF A$="■" THEN X=X-1:CURSORX,Y GOTO510
560 IF A$="■" THEN Y=Y+1:CURSORX,Y:GOTO510
570 IF A$="■" THEN X=X-1:CURSORX,Y GOTO510
580 IF A$="A" THEN K1=K:K=15:GOTO620
590 IF A$="E" THEN K=15:J1=J-1:J=30:GOTO630
600 IF A$<>"Z" THEN510 ELSE 610
610 W(U)=X:W(U+1)=20-Y:U=U+2:CURSOR22+F,JJ*2+1:PRINT"#":CURSORX,Y:PRINT"*":F=F+1
620 :NEXTK:U1=U-F*2-1:W(U1)=F:F=0
630 NEXTJ:CURSOR22,17:PRINT"*ｶｸｽｳ";J1:W(1)=J1:PRINTCHR$(26)
640 PRINT GYU;"DATA ";NA$;
650 FOR N=1TO U-2
660 IF W(N)>9 THEN PRINTUSING"_,##";W(N); ELSE PRINTUSING"_,#";W(N);
670 NEXT
680 END
690 'ｴﾗｰ ｼｮﾘ
700 CONSOLE 0,17,22,17:CLS:CONSOLE0,25,0,40:CURSOR22,0:F=0:BEEP
710 IF U=0 THEN X=0:Y=0:RESUME ELSE PRINT"ﾔﾘﾅｵｼ":RESUME 450
720 '**ｶﾝｼﾞ ｦ ｶｸ**
730 MODE GR:HSET:SKL=1:GOSUB 740:END
740 READ NAMAE$,KAKU:PRINT NAMAE$;KAKU
750 FOR NI=1 TO KAKU
760 READ TEN,YOKO,TATE:MOVE YOKO*SKL,TATE*SKL:PRINTTEN;
770 FOR NJ=1 TO TEN-1
780 READ YOKO,TATE:LINE YOKO*SKL,TATE*SKL
790 NEXTNJ:NEXTNI:PRINT
800 PHOME:MOVE24,0:MODE TN:RETURN
810 '**ｶｸﾀﾞｲ ｼﾃ ｶｸ**
820 PCOLOR1:SKL=5:SKIP5:MODEGR
830 FOR I=0TO4:LINEI*25,0,I*25,100:MOVE(I+1)*25,0:NEXT:MOVE0,0
840 FOR I=0TO4:LINE0,I*25,100,I*25:MOVE0,(I+1)*25:NEXT:PHOME
850 PCOLOR0:GOSUB740
860 END
1000 'DATA ﾋﾄ,2,3,10,17,9,9,2,0,3,9,9,15,0,18,0
2000 DATA ｶｽﾞ,13,2,3,18,4,16,2,8,16,10,18,2,10,15,2,15,2,6,18,6,10,2,2,10,4,13,2
,8,13,10,10,2,10,7,2,7,3,6,9,2,2,8,0,3,8,6,7,2,3,0,3,11,1,17,8,18,15,2,19,15,13,
15,2,13,18,11,13,4,13,13,13,8,17,0,19,0
```

リスト3

```
100 '***ﾌﾟﾛｸﾞﾗﾑ ﾚｲ***
110 KSTI=0
120 FOR I=1 TO 2
130 KAZ=2:KN(1)=1:KN(2)=2:GOSUB"ｶﾝｼﾞ"
140 LPRINT"ﾊ､";
150 KAZ=2:KN(1)=4:KN(2)=5:GOSUB"ｶﾝｼﾞ"
160 LPRINT"ｶﾞ";
170 KAZ=1:KN(1)=3:GOSUB"ｶﾝｼﾞ"
180 LPRINT"ｲﾃﾞｽ｡":RET=0
190 NEXT I:END
3000 '***ｶﾝｼﾞ ﾉ ｻﾌﾞﾙｰﾁﾝ***
3010 LABEL"ｶﾝｼﾞ"
3020 SKL=KSTI+1
3030 IF RET=0 THEN LPS=0:KNK=0
3040 RET=RET+1
3050 LPS=LPS+LPOS(0)
3060 PGS=KNK+LPS
3070 LPRINT:MODE GR:MOVE PGS*12,24:HSET
3080 FOR NW=1 TO KAZ
3090 HSET
3100 ON KN(NW) RESTORE 5000,5100,5200,5300,5400
3110 IF KN(NW)=0 THEN 3180
3120 READ NAMAE$,KAKU
3130 FOR NI=1 TO KAKU
3140 READ TEN,YOKO,TATE:MOVE YOKO*SKL,TATE*SKL
3150 FOR NJ=1 TO TEN-1
3160 READ YOKO,TATE:LINE YOKO*SKL,TATE*SKL
3170 NEXT NJ,NI
3180 PHOME:MOVE 24*SKL,0
3190 NEXT NW:MODE TN
3200 KNK=KNK+KAZ*2*SKL:RETURN
5000 DATA ｲﾏ,3,4,1,9,10,17,17,10,19,9,2,14,11,7,11,5,6,8,15,8,13,4,9,1,6,0
5100 DATA ﾆﾁ,2,6,3,0,3,15,15,15,15,8,15,0,3,0,2,3,8,15,8
5200 DATA ﾖｲ,5,6,8,1,6,0,6,14,14,14,14,8,6,8,2,6,11,14,11,2,10,16,10,14,2,14,6,1
0,2,3,10,6,13,2,16,0
5300 DATA ﾃﾝ,4,3,3,1,9,7,10,15,2,4,15,16,15,2,15,10,5,10,3,9,7,13,3,17,1
5400 DATA ｷ,6,3,4,1,8,4,11,7,2,6,6,10,2,6,18,2,18,0,14,2,14,6,15,10,5,10,2,6,12,
13,12,2,14,14,5,14,2,6,16,3,12
```

リスト2

```
3000 '***ｶﾝｼﾞ ﾉ ｻﾌﾞﾙｰﾁﾝ***
3010 LABEL"ｶﾝｼﾞ"
3020 SKL=KSTI+1
3030 IF RET=0 THEN LPS=0:KNK=0
3040 RET=RET+1
3050 LPS=LPS+LPOS(0)
3060 PGS=KNK+LPS
3070 LPRINT:MODE GR:MOVE PGS*12,24
3080 FOR NW=1 TO KAZ
3090 HSET
3100 ON KN(NW) RESTORE
3110 IF KN(NW)=0 THEN 3180
3120 READ NAMAE$,KAKU
3130 FOR NI=1 TO KAKU
3140 READ TEN,YOKO,TATE:MOVE YOKO*SKL,TATE*SKL
3150 FOR NJ=1 TO TEN-1
3160 READ YOKO,TATE:LINE YOKO*SKL,TATE*SKL
3170 NEXT NJ,NI
3180 PHOME:MOVE 24*SKL,0
3190 NEXT NW:MODE TN
3200 KNK=KNK+KAZ*2*SKL:RETURN
```

MZ-2200/2000(GRAM1)

ゲーム　MONKEY THREE

—オールマシン語—

笠原　千秋

プログラム説明

マシン語に早く慣れるつもりで，１週間かかって完成させたゲームです。私自身マシン語はほんの初心者なので，アルゴリズムは比較的やさしく，内容はイマイチといったところです。それでもグラフィック・ルーチンや乱数発生ルーチンなどで，何回となくZ80-CPUの拒絶反応に遭いました。

シャープのシステムプログラムではなく，別なアセンブラを使って組み立てたので，ソースリストは発表いたしません。

ゲーム方法は，３匹のMONKEYが木から木へ飛び移って木の実を落としてきます。MANは木の実に当たらないようにして，猿を弓矢で射落としてください。

キーの操作は，

[1]　：LEFT
[3]　：RIGHT
[5]　：JUMP
[SPACE]：SHOT

MANが３人やられるとGAME OVERとなり，モニタに戻ります。再びゲームを始めるときは，いちど画面をクリアして＊J [CR] J-ADR. ＄8000 [CR] としてください。

モニタのサウンドルーチンの使い方がまだよくわからないので，効果音には３オクターブのメロディルーチンだけ使っています。

V-RAMの画面を初めて決めてしまったために，９ラウンド以上をクリアすると，０ラウンドに戻ります（SCOREはそのまま増えていく）。

モニタ・サブルーチンのCALLMSGは，V-RAMなどとクロスさせて使用できませんが，文字列の表示には便利なので，V-RAMをオフにして使ってみました。マシン語に不慣れなので，乱数を使っても，猿の動きは単調ですが，３匹のうち，どの猿に矢が当たったかをグラフィック・ルーチンで判定するのに相当苦労しました。



マシン語の入力方法

まずMZ-1Z 001をロードします。

MON [CR]
＊M [CR]
M-ADR. ＄8000 [CR]

として，リストどおり打ち込みます。

チェックサムを確認したなら，

＊S [CR]
FILE NAME：MONKEY THREE [CR]
S-ADR. ＄8000 [CR]
E-ADR. ＄8AF3 [CR]
J-ADR. ＄8000 [CR]

としてセーブします。次回からは，ロード後にオートスタートします。

なお，このゲームはグラフィックRAM1ページを使っています。

以下にマシン語ルーチンについての処理内容を簡単に説明します。

処理内容

ORG 8000H	
8003～8005	乱数ワークエリア
8006～800A	スコア表示ワークエリア
800B～802A	人，猿，木の実のワークエリア
802B～803A	VRAM，GRAM　オン
803B～8056	画面クリア
8057～80DE	背景設定
80DF～80E3	VRAM，GRAM　オフ
80E4～826C	人，猿のグラフィック・データ
826D～82D3	矢，木の実のグラフィック
82D4～82F2	乱数発生
82F3～830C	タイムデイレイ
830D～8449	タイトル
844A～84C3	初期設定
84C4～866E	メインルーチン
866F～8714	木の実動かす
8715～8727	人やられた
8728～87CC	キー操作
87CD～885C	猿に当たる
885D～891E	スコア表示
891F～8980	ラウンドクリア
8981～89B1	GAME OVER
89B2～89D3	JUMP判定
89D4～8AF3	音楽データ

リスト

```
** Oh!MZ Check sum system V1.0 **

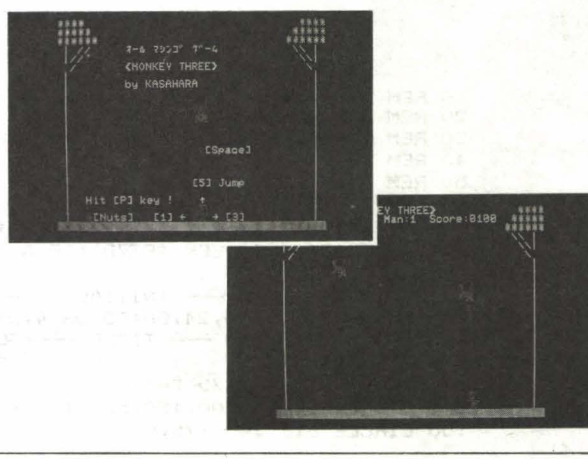
ADR. +0 +1 +2 +3 +4 +5 +6 +7 :sum.
8000 C3 4A 84 35 00 34 30 30 :2.5A
8008 30 30 03 03 DE 01 B5 D6 :2.D0
8010 01 14 CF 01 28 F7 30 31 :2.65
8018 80 C7 CE C7 00 00 00 00 :2.DC
8020 F7 00 DE 80 D6 00 CF A9 :4.A3
8028 F9 84 F9 3E C0 D3 E8 C9 :5.F8
8030 3E 80 D3 E8 3E 01 D3 F7 :4.82
8038 D3 F6 C9 01 E8 03 11 01 :3.90
----------------------------------1D18
8040 D0 21 00 D0 AF 77 ED B0 :4.84
8048 C9 01 80 3E 11 01 C0 21 :2.7B
8050 00 C0 AF 77 ED B0 C9 CD :5.19
8058 2B 80 CD 3B 80 21 00 D0 :3.24
8060 11 20 00 06 04 3E 2A 77 :1.1A
8068 23 10 FC 19 06 09 77 23 :1.F1
8070 10 FC 11 1E 00 19 06 0B :1.65
8078 77 23 10 FC 11 1C 00 19 :1.EC
----------------------------------1598
8080 06 06 77 23 10 FC 23 3E :2.13
8088 9A 77 23 23 3E 2F 77 23 :2.5E
8090 77 11 1F 00 19 3E 5C 77 :1.D1
8098 23 77 23 23 3E 9A 77 23 :2.52
80A0 23 23 77 23 3E 2F 77 23 :1.E7
80A8 77 11 21 00 19 3E 5C 77 :1.D3
80B0 23 77 23 3E 9A 77 23 23 :2.52
80B8 23 23 3E 2F 77 11 23 00 :1.5E
----------------------------------0FFE
80C0 19 3E 5C 77 23 23 23 06 :1.99
80C8 12 11 25 00 3E 9A 23 77 :1.BA
80D0 19 77 23 23 10 F8 06 28 :2.0C
80D8 3E 1F 77 23 10 FC C9 3E :3.0A
80E0 50 D3 E8 C9 DD 21 2D 81 :4.80
80E8 CD 0C 81 C9 DD 21 6D 81 :4.0F
80F0 CD 0C 81 C9 DD 21 AD 81 :4.4F
80F8 CD 0C 81 C9 DD 21 ED 81 :4.8F
----------------------------------19D6
8100 CD 0C 81 C9 DD 21 2D 82 :3.D0
8108 CD 0C 81 C9 C5 D5 E5 11 :4.B3
8110 50 00 0E 04 06 10 DD 7E :1.D3
8118 00 77 DD 23 19 10 F7 E1 :3.78
8120 23 E5 0D 20 EF E1 2B 2B :3.5B
8128 2B 2B D1 C1 C9 00 00 00 :2.B1
8130 00 00 00 00 00 00 00 00 :0.00
8138 00 00 00 00 00 00 00 00 :0.00
----------------------------------13DA
8140 00 00 00 00 00 00 00 00 :0.00
8148 00 00 00 00 00 00 00 00 :0.00
8150 00 00 00 00 00 00 00 00 :0.00
8158 00 00 00 00 00 00 00 00 :0.00
8160 00 00 00 00 00 00 00 00 :0.00
8168 00 00 00 00 00 80 40 20 :0.E0
8170 10 08 04 02 01 FF 00 00 :1.1E
8178 00 00 00 80 80 FF FF 3C :3.3A
----------------------------------0538
8180 CC E6 E6 EE DE FF F8 F0 :7.4B
8188 E0 E0 F0 3F 1F FF FF 00 :5.0C
8190 03 07 07 07 03 FF 1F 3F :1.7B
8198 F7 E7 0F FE FC 01 02 04 :3.EE
81A0 08 10 20 40 80 FF 3E 03 :2.38
81A8 01 00 00 01 01 38 7C C6 :1.7D
81B0 83 03 03 03 00 00 F0 F8 :2.74
81B8 F8 18 18 30 7F CC F4 E8 :4.7F
----------------------------------1C65
81C0 91 F3 76 EC 5C FC FC 9D :5.D7
81C8 1F 3F FE F0 00 3B 2F 1F :2.D5
81D0 19 9F 9D CF E4 7F 3F B9 :4.7F
81D8 F8 FC 7F 0F 00 1C 3E 33 :3.0F
81E0 63 C1 C1 C0 00 00 0F 1F :2.D3
81E8 1F 18 18 0C 7E 1E 31 61 :1.89
81F0 61 61 32 98 CC FE FE FE :5.52
81F8 FC F8 FC 7E 03 00 00 00 :3.71
----------------------------------1D59
8200 00 00 00 FF FF FF FF 0F :4.0B
8208 FF 1F 3E 7C E0 80 C0 80 :4.78
8210 80 80 C0 FF FF FF FF 00 :5.BC
8218 FF 00 00 00 03 7D FE 9F :3.1C
8220 F7 17 77 3F 03 87 CF FF :4.1C
8228 FC 78 30 00 00 BE 7F F9 :3.DA
8230 EF E8 EE FC E0 E1 F3 FF :7.74
8238 3F 1E 0C 00 00 01 03 01 :0.6E
----------------------------------2133
8240 01 01 03 FF FF FF FF 01 :4.02
8248 FF 00 00 00 C0 00 00 00 :1.BF
8250 00 00 00 FF FF FF FF F0 :4.EC
8258 FF F8 7C 3E 07 78 8C 86 :4.42
8260 86 86 4C 19 33 7F 7F 7F :3.21
8268 3F 1F 3F 7E C0 DD 21 A4 :3.7D
8270 82 CD 85 82 C9 DD 21 B4 :4.D1
8278 82 CD 85 82 C9 DD 21 C4 :4.E1
----------------------------------1F3F
8280 82 CD 85 82 C9 C5 D5 E5 :5.9E
8288 11 50 00 0E 02 06 08 DD :1.5C
8290 7E 00 77 DD 23 19 10 F7 :3.15
8298 E1 23 E5 0D 20 EF E1 2B :3.11
82A0 2B D1 C1 C9 C0 A0 90 80 :4.F6
82A8 80 E0 10 10 03 05 09 01 :1.92
82B0 01 07 08 08 F0 A8 D4 FF :3.83
82B8 FF D4 A8 F0 0F 15 2B FF :4.B9
----------------------------------1CE4
82C0 FF 2B 15 0F 00 00 00 00 :1.4E
82C8 00 00 00 00 00 00 00 00 :0.00
82D0 00 00 00 00 FD E5 FD 21 :3.00
82D8 03 80 3A 05 80 47 87 87 :2.97
82E0 4F 87 80 81 3C 32 05 80 :2.CA
82E8 FE 3C 30 EE 3C FD 77 00 :4.08
82F0 FD E1 C9 C5 01 F4 01 0B :4.6D
82F8 78 B1 FE 00 20 F9 C1 C9 :4.CA
----------------------------------16EE
8300 CD F3 82 CD F3 82 CD F3 :6.44
8308 82 CD F3 82 C9 CD 57 80 :5.31
8310 11 81 83 CD 89 08 CD 2E :3.6E
8318 0A 11 9E 83 CD 89 08 CD :3.67
8320 2E 0A 11 B8 83 CD 89 08 :2.E2
8328 CD 2E 0A 11 CF 83 CD 89 :3.BE
8330 08 CD 2E 0A 11 F4 83 CD :3.62
8338 89 08 CD 2E 0A 11 14 84 :2.3F
----------------------------------1E8B
8340 CD 89 08 CD 2E 0A 11 2C :2.A0
8348 84 CD 89 08 CD 30 80 CD :4.2C
8350 49 80 21 29 D9 CD F4 80 :4.2D
8358 21 AA E5 CD 6D 82 21 B8 :4.15
8360 F9 CD 75 82 21 29 F7 CD :4.CB
8368 EC 80 CD D4 89 CD 32 08 :4.9D
8370 FE 50 20 F9 CD 30 80 CD :4.B1
8378 49 80 CD 2B 80 CD 3B 80 :3.C9
----------------------------------20F0
8380 C9 05 01 01 01 03 03 03 :0.DA
```

```
8388 03 03 03 03 03 03 03 B5 :0.CA
8390 2D D9 20 CF BC DD BA DE :5.26
8398 20 B9 DE 2D D1 0D 01 03 :2.C6
83A0 03 03 03 03 03 03 03 03 :0.18
83A8 03 3C 4D 4F 4E 4B 45 59 :2.12
83B0 20 54 48 52 45 45 3E 0D :1.E3
83B8 01 03 03 03 03 03 03 03 :0.16
-------------------------------------0DB3
83C0 03 03 03 62 79 20 4B 41 :1.90
83C8 53 41 48 41 52 41 0D 01 :1.BE
83D0 01 01 01 01 01 01 03 03 :0.0C
83D8 03 03 03 03 03 03 03 03 :0.18
83E0 03 03 03 03 03 03 03 03 :0.18
83E8 03 03 03 03 5B 53 70 61 :1.8B
83F0 63 65 5D 0D 01 01 01 03 :1.38
83F8 03 03 03 03 03 03 03 03 :0.18
-------------------------------------0665
8400 03 03 03 03 03 03 03 03 :0.18
8408 03 03 03 5B 35 5D 20 4A :1.60
8410 75 6D 70 0D 01 03 03 03 :1.69
8418 03 48 69 74 20 5B 50 5D :2.50
8420 20 6B 65 79 20 21 03 03 :1.B0
8428 03 03 82 0D 01 03 03 03 :0.9F
8430 03 03 5B 4E 75 74 73 5D :2.68
8438 03 03 03 5B 31 5D 20 84 :1.96
-------------------------------------0B7E
8440 20 20 20 20 83 20 5B 33 :1.B1
8448 5D 0D CD 0D 83 DD 21 06 :2.CB
8450 80 3E 30 DD 77 00 DD 77 :3.96
8458 01 DD 77 02 DD 77 03 3E :2.EC
8460 31 32 17 80 3E 33 32 16 :1.B3
8468 80 DD 21 02 DE DD 22 0B :3.68
8470 80 DD 21 82 D6 DD 22 0E :3.E3
8478 80 DD 21 02 CF DD 22 11 :3.5F
-------------------------------------175B
8480 80 3E 01 32 0D 80 32 10 :1.C0
8488 80 32 13 80 3E 03 32 0A :1.C2
8490 80 CD D4 82 ED 5B 03 80 :4.6E
8498 2A 0B 80 19 22 0B 80 CD :2.48
84A0 D4 82 ED 5B 03 80 2A 0E :3.59
84A8 80 19 22 0E 80 CD D4 82 :3.6C
84B0 ED 5B 03 80 2A 11 80 19 :2.9F
84B8 22 11 80 21 29 F7 22 14 :2.2A
-------------------------------------15C6
84C0 80 CD 57 80 CD 5D 88 3A :4.10
84C8 16 80 FE 31 DA 81 89 3A :3.E3
84D0 0A 80 FE 01 DA 1F 89 CD :3.D8
84D8 B2 89 3A 0D 80 FE 00 28 :3.28
84E0 39 CD D4 82 3A 03 80 47 :3.60
84E8 2A 0B 80 CD FC 80 22 0B :3.2B
84F0 80 CD 28 87 CD 00 83 CD :4.19
84F8 E4 80 3A 0D 80 FE 00 28 :3.51
-------------------------------------1CE8
8500 19 7D FE 49 28 0B 23 10 :2.43
8508 E2 CD FC 80 22 0B 80 18 :3.F0
8510 09 CD FC 80 22 0B 80 CD :3.CC
8518 6F 86 CD B2 89 3A 10 80 :3.C7
8520 FE 00 28 39 CD D4 82 3A :3.BC
8528 03 80 47 2A 0E 80 CD FC :3.4B
8530 80 22 0E 80 CD 28 87 CD :3.79
8538 00 83 CD E4 80 3A 10 80 :3.7E
-------------------------------------1BC4
8540 FE 00 28 19 7D FE C9 28 :3.AB
8548 0B 23 10 E2 CD FC 80 22 :3.8B
8550 0E 80 18 09 CD FC 80 22 :3.1A
8558 0E 80 CD 6F 86 CD B2 89 :4.58
8560 3A 13 80 FE 00 28 39 CD :2.F9
8568 D4 82 3A 03 80 47 2A 11 :2.95
8570 80 CD FC 80 22 11 80 CD :4.49
8578 28 87 CD 00 83 CD E4 80 :4.30
-------------------------------------1CAF
8580 3A 13 80 FE 00 28 19 7D :2.89
```

```
8588 FE 49 28 0B 23 10 E2 CD :3.5C
8590 FC 80 22 11 80 18 09 CD :3.1D
8598 FC 80 22 11 80 CD 6F 86 :3.F1
85A0 CD B2 89 3A 0D 80 FE 00 :3.CD
85A8 28 39 CD D4 82 3A 03 80 :3.41
85B0 47 2A 0B 80 CD 04 81 22 :2.70
85B8 0B 80 CD 28 87 CD 00 83 :3.57
-------------------------------------19C8
85C0 CD E4 80 3A 0D 80 FE 00 :3.F6
85C8 28 19 7D FE 03 28 0B 2B :2.1D
85D0 10 E2 CD 04 81 22 0B 80 :2.F1
85D8 18 09 CD 04 81 22 0B 80 :2.20
85E0 CD C2 86 CD B2 89 3A 10 :4.67
85E8 80 FE 00 28 39 CD D4 82 :4.02
85F0 3A 03 80 47 2A 0E 80 CD :2.89
85F8 04 81 22 0E 80 CD 28 87 :2.B1
-------------------------------------18C7
8600 CD 00 83 CD E4 80 3A 10 :3.CB
8608 80 FE 00 28 19 7D FE 83 :3.BD
8610 28 0B 2B 10 E2 CD 04 81 :2.A2
8618 22 0E 80 18 09 CD 04 81 :2.23
8620 22 0E 80 CD C2 86 CD B2 :4.44
8628 89 3A 13 80 FE 00 CA C4 :3.E2
8630 84 CD D4 82 3A 03 80 47 :3.AB
8638 2A 11 80 CD 04 81 22 11 :2.40
-------------------------------------1A5E
8640 80 CD 28 87 CD 00 83 CD :4.19
8648 E4 80 3A 13 80 FE 00 CA :3.F9
8650 C4 84 7D FE 03 28 0C 2B :3.25
8658 10 E1 CD 04 81 22 11 80 :2.F6
8660 C3 C4 84 CD 04 81 22 11 :3.90
8668 80 CD C2 86 C3 C4 84 CD :5.6D
8670 A8 8A 2A 1A 80 11 40 01 :2.48
8678 CD 75 82 CD F3 82 CD 7D :5.50
-------------------------------------1EC2
8680 82 19 EB 21 CE F9 ED 52 :4.AD
8688 28 03 EB 18 E8 EB 22 29 :3.4C
8690 80 2A 29 80 CD 75 82 CD :3.E4
8698 06 83 CD 7D 82 2B 7D FE :3.FB
86A0 83 C8 22 29 80 CD 28 87 :3.92
86A8 11 7F 02 ED 52 7E FE 00 :3.4D
86B0 28 DF 2A 29 80 2B CD 75 :3.47
86B8 82 CD D2 8A CD 7D 82 C3 :5.3A
-------------------------------------1F38
86C0 15 87 CD A8 8A 2A 18 80 :3.5D
86C8 11 40 01 CD 75 82 CD F3 :3.D6
86D0 82 CD 7D 82 19 EB 21 80 :3.F3
86D8 F9 ED 52 28 03 EB 18 E8 :4.4E
86E0 EB 22 27 80 2A 27 80 CD :3.52
86E8 75 82 CD 06 83 CD 7D 82 :4.19
86F0 23 7D FE CC C8 22 27 80 :3.FB
86F8 CD 28 87 11 80 02 ED 52 :3.4E
-------------------------------------1E28
8700 7E FE 00 28 DF 2A 27 80 :3.54
8708 23 CD 75 82 CD D2 8A CD :4.DD
8710 7D 82 C3 15 87 2A 14 80 :3.1C
8718 CD E4 80 CD D6 88 CD 5D :5.86
8720 88 21 28 F7 22 14 80 C9 :3.47
8728 D9 2A 14 80 CD EC 80 CD :4.9D
8730 34 87 D9 C9 3E FF 32 F5 :4.C1
8738 11 32 F6 11 CD 32 08 FE :3.4F
-------------------------------------20C7
8740 31 28 0D FE 33 28 1E FE :2.DB
8748 35 28 2F FE 20 28 4C C9 :2.E7
8750 3A 1E 80 FE 01 C8 7D FE :4.1A
8758 07 D8 CD E4 80 2B CD EC :4.F4
8760 80 22 14 80 C9 3A 1E 80 :2.D7
8768 FE 01 C8 7D FE 44 D0 CD :5.23
8770 E4 80 23 CD EC 80 22 14 :3.F6
8778 80 C9 3A 1E 80 FE 01 C8 :3.E8
-------------------------------------1EA8
8780 06 02 11 80 02 CD E4 80 :2.CC
```

```
8788 ED 52 CD EC 80 CD 06 83 :4.CE
8790 10 F3 22 14 80 3E 01 32 :2.2A
8798 1E 80 C9 3A 1E 80 FE 01 :3.3E
87A0 C8 2A 14 80 11 80 02 ED :3.06
87A8 52 23 CD 95 8A 11 40 01 :2.B3
87B0 CD 6D 82 CD 00 83 CD 7D :4.56
87B8 82 22 1C 80 ED 4B 25 80 :3.1D
-----------------------------------1A2E
87C0 ED 42 F8 2A 1C 80 ED 52 :4.2C
87C8 7E FE 00 28 E3 CD 6D 82 :4.43
87D0 CD 00 83 CD 7D 82 22 1C :3.5A
87D8 80 ED 4B 21 80 ED 42 F2 :4.7A
87E0 01 88 2A 1C 80 ED 4B 23 :2.AA
87E8 80 ED 42 F2 15 88 2A 1C :3.84
87F0 80 ED 4B 25 80 ED 42 F2 :4.7E
87F8 29 88 CD DE 88 CD 5D 88 :4.96
-----------------------------------1F85
8800 C9 2A 0B 80 CD 3D 88 AF :3.BF
8808 32 0D 80 3A 0A 80 3D 32 :1.F2
8810 0A 80 C3 FA 87 2A 0E 80 :3.86
8818 CD 3D 88 AF 32 10 80 3A :3.3D
8820 0A 80 3D 32 0A 80 C3 FA :3.40
8828 87 2A 11 80 CD 3D 88 AF :3.83
8830 32 13 80 3A 0A 80 3D 32 :1.F8
8838 0A 80 C3 FA 87 06 03 11 :2.E8
-----------------------------------1817
8840 80 02 CD F4 80 CD 00 83 :4.13
8848 CD E4 80 ED 52 CD F4 80 :5.B1
8850 CD BD 8A CD E4 80 19 CD :5.2B
8858 00 83 10 E6 C9 CD DF 80 :4.6E
8860 11 9E 88 CD 89 08 CD 2E :3.90
8868 0A 11 BB 88 CD 89 08 CD :3.86
8870 2B 80 3A 17 80 32 35 D0 :2.B3
8878 3A 16 80 32 3C D0 DD 21 :3.0C
-----------------------------------2032
8880 06 80 DD 7E 00 32 45 D0 :3.2B
8888 DD 7E 01 32 46 D0 DD 7E :3.FF
8890 02 32 47 D0 DD 7E 03 32 :2.DB
8898 48 D0 CD 30 80 C9 05 03 :3.66
88A0 03 03 03 03 03 03 03 03 :0.18
88A8 03 3C 4D 4F 4E 4B 45 59 :2.12
88B0 20 54 48 52 45 45 3E 0D :1.E3
88B8 03 03 03 03 03 03 03 52 :0.67
-----------------------------------11DC
88C0 6F 75 6E 64 3A 03 03 03 :1.F9
88C8 4D 61 6E 3A 03 03 03 53 :1.B2
88D0 63 6F 72 65 3A 0D 3A 16 :2.40
88D8 80 3D 32 16 80 C9 DD 21 :3.4C
88E0 06 80 DD 7E 02 FE 39 28 :3.42
88E8 05 3C DD 77 02 C9 3E 30 :2.CE
88F0 DD 77 02 DD 7E 01 FE 39 :3.E9
88F8 28 05 3C DD 77 01 C9 3E :2.C5
-----------------------------------15F5
8900 30 DD 77 01 DD 7E 00 FE :3.DE
8908 39 28 05 3C DD 77 00 C9 :2.BF
8910 3E 30 DD 77 00 DD 77 01 :3.17
8918 DD 77 02 DD 77 03 C9 CD :4.43
8920 29 8A 3A 17 80 FE 39 28 :2.E3
8928 06 3C 32 17 80 18 05 3E :1.66
8930 30 32 17 80 CD DF 80 11 :3.36
8938 53 89 CD 89 08 CD 32 08 :3.41
-----------------------------------18B7
8940 FE 50 20 F9 CD 30 80 CD :4.B1
8948 49 80 CD 2B 80 CD 3B 80 :3.C9
8950 C3 69 84 05 01 01 01 01 :1.B9
8958 01 01 01 01 01 03 03 03 :0.0E
8960 03 03 03 03 52 4F 55 4E :1.50
8968 44 43 4C 45 41 52 20 20 :1.EB
8970 83 83 83 48 69 74 20 5B :3.29
8978 50 5D 20 6B 65 79 20 21 :2.57
-----------------------------------12FC
```

```
8980 0D CD 66 8A CD DF 80 11 :4.07
8988 90 89 CD 89 08 C3 B1 00 :3.EB
8990 05 01 01 01 01 01 01 01 :0.0C
8998 01 01 03 03 03 03 03 03 :0.14
89A0 03 03 03 03 47 41 4D 45 :1.26
89A8 20 4F 56 45 52 20 21 21 :1.BE
89B0 21 0D 2A 14 80 CD EC 80 :3.25
89B8 3A 1E 80 FE 00 C8 06 02 :2.A6
-----------------------------------10C1
89C0 11 80 02 CD E4 80 19 CD :3.AA
89C8 F3 82 10 F7 22 14 80 AF :3.E1
89D0 32 1E 80 C9 3E 06 CD 50 :2.FA
89D8 0E 11 E0 89 CD 3F 0F C9 :3.6C
89E0 46 31 47 31 46 33 46 33 :1.E1
89E8 46 33 46 35 52 31 42 33 :1.EC
89F0 2B 43 33 2B 44 33 2B 43 :1.B1
89F8 31 46 31 46 31 41 33 41 :1.D4
-----------------------------------1543
8A00 33 41 33 41 37 52 31 42 :1.E4
8A08 31 42 31 42 31 2B 43 31 :1.B6
8A10 42 33 42 33 42 33 42 35 :1.D6
8A18 52 31 45 33 45 33 46 33 :1.EC
8A20 46 33 46 33 46 37 52 31 :1.F2
8A28 0D 3E 06 CD 50 0E 11 35 :1.C2
8A30 8A CD 3F 0F C9 47 31 47 :3.2D
8A38 31 2B 43 31 2B 43 31 2B :1.9A
-----------------------------------0FD7
8A40 44 31 2B 44 31 2B 45 31 :1.B6
8A48 2B 45 31 2B 45 31 2B 44 :1.B1
8A50 31 2B 44 31 2B 43 31 2B :1.9B
8A58 43 31 2B 44 31 2B 44 31 :1.B4
8A60 2B 45 37 52 31 0D 3E 06 :1.7B
8A68 CD 50 0E 11 72 8A CD 3F :3.44
8A70 0F C9 2D 43 31 2D 44 31 :2.1B
8A78 2D 46 31 2D 42 31 46 31 :1.BB
-----------------------------------0F4B
8A80 2B 43 31 2B 23 42 31 2B :1.8B
8A88 44 31 42 31 44 31 2D 42 :1.CC
8A90 31 2D 47 33 0D D5 3E 06 :1.FE
8A98 CD 50 0E 11 A3 8A CD 3F :3.75
8AA0 0F D1 C9 43 30 45 31 0D :2.9F
8AA8 D5 3E 06 CD 50 0E 11 B6 :3.0B
8AB0 8A CD 3F 0F D1 C9 2D 42 :3.AE
8AB8 32 2D 46 32 0D D5 3E 06 :1.FD
-----------------------------------141F
8AC0 CD 50 0E 11 CB 8A CD 3F :3.9D
8AC8 0F D1 C9 2B 46 31 2B 44 :2.BA
8AD0 31 0D D5 3E 06 CD 50 0E :2.82
8AD8 11 E0 8A CD 3F 0F D1 C9 :4.30
8AE0 2D 41 31 2D 41 31 2D 42 :1.AD
8AE8 31 2D 42 31 2D 41 31 2D :1.9D
8AF0 41 31 0D 00 00 00 00 00 :0.7F
8AF8 00 00 00 00 00 00 00 00 :0.00
-----------------------------------10D2
```



UFO撃墜ゲーム

MZ-2200／2000 (G-RAM, I ,II,III)

カラーBASIC＋マシン語

辻 賢司

ゲームの遊び方

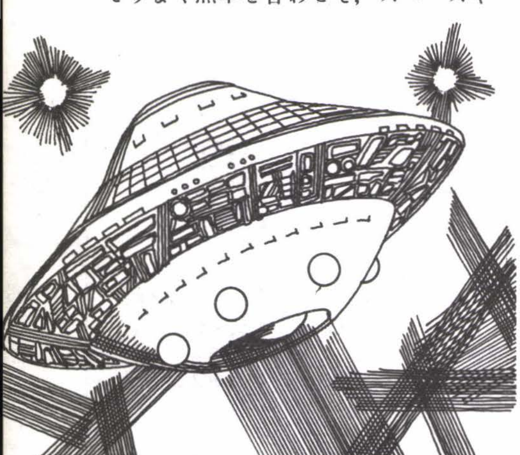
RUN [CR] で，キャラクタパターンデータの入ったマシン語部分を読み込み，オートスタートします。

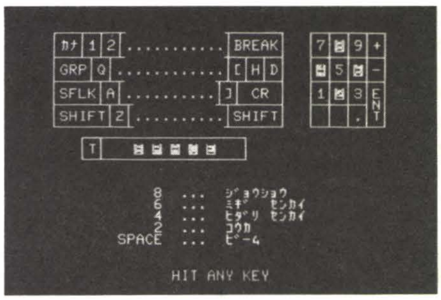
タイトル画面のあとに，キー操作の説明があります。テンキーの2･4･6･8とSPACEキーで操作するようになっています。

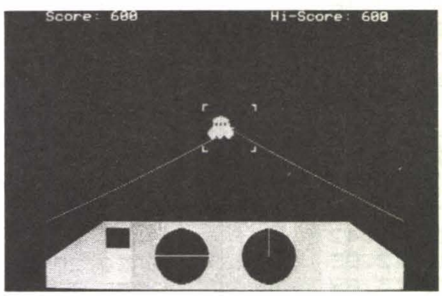
UFOが発見されるとレーダーに映ります。テンキーの4・6がHレーダーで方向，2・8がVレーダーで高度を合わせると，前方スクリーンにUFOが現われますので，テンキーでうまく照準を合わせて，スペースキーでビームを発射してくだかい。

UFOの真ん中に命中すれば200点，真ん中でなくてもUFOが破壊されれば100点の得点が与えられます。しかし，ビームを撃つごとにエネルギーが消耗し，エネルギーが0になるとビームが発射できなくなりますので，むだ射ちに注意してください。

UFOから3回攻撃されるとGAME OVERです。







プログラムについて

マシン語DATAは＄F37B～＄FA00です。Sコマンドで，

FILE NAME：UFO DATA [CR]
S-ADR. $ F37B [CR]
E-ADR. $ FA00 [CR]
J-ADR. $ F952 [CR]

で，BASIC部分をSAVE後，続けてSAVEしてください。

このキャラクタパターンデータは，アーバン電子の「カラーパターンエディタ」で作成しました。これはドット単位で作成するので，小さく，たいへん見にくいのですが，カラーが36色使え，ドット単位にカラーを設定できるので，ゲーム作りに活用しています。



リスト1 BASICプログラム

```
10 REM *******************************
20 REM *                             *
30 REM *  UFO ゲキツイ ゲーム  BY JR2HIN  *
40 REM *                             *
50 REM *       1983/05/06            *
60 REM *                             *
70 REM *******************************
80 LIMIT $9000:A=PEEK($F37B):IF A<>25 THEN LOAD "UFO DATA":REM ―― DATA LOAD ――
90 USR($F8CD):REM ―― INITIALIZE ――
100 CONSOLE C40,S0,24,GH:COLOR 4,07,W0:TEMPO 7
110 GOSUB 1310:REM ―― TITLE ――:GOSUB 1570:REM ―― ゲーム セツメイ ――
120 PRINT CHR$(6)
130 REM ―― ショキ ガメン セッテイ ――
140 LINE[1]0,180,100,150,539,150,639,180
150 CIRCLE [1] 240,175,50
```

```
 160 CIRCLE [1] 240,175,51:LINE [6] 190,175,290,175
 170 CIRCLE [1] 400,175,50
 180 CIRCLE [1] 400,175,51:LINE [6] 400,155,400,175
 190 BOX [1] 100,152,150,198,F1:BOX [6] 105,155,145,195,F6
 200 BOX [1] 489,152,539,198,F1:BOX 494,155,534,195,F5
 210 PAINT [4] 100,151,1
 220 CURSOR 17, 8:PRINT "┌      ┐"
 230 CURSOR 17,10:PRINT "   +   "
 240 CURSOR 17,12:PRINT "└      ┘"
 250 CURSOR 0,0:PRINT"Score:";SC:CURSOR 25,0:PRINT"Hi-Score:";HI
 260 CURSOR 12,19:PRINT"U           L     R"
 270 CURSOR 0,23:PRINT "ENERGY                V  H            DAMAGE";
 280 CURSOR 12,24:PRINT"D";
 290 POKE1140,175:POKE1141,195:POKE1142,1:POKE1143,9
 300 REM ―― UFO ﾊｯｾｲ ――
 310 FOR I=0 TO 1000:NEXT:R=30:B=0:LINE [6] 190,175,290,175:LINE [6] 400,155,400
,175
 320 MUSIC"+C1+D+C+D":TI$="000000":A=INT(RND(1)*36)*10:GOSUB 1150:IF A=270 THEN
H=1
 330 UV=INT(RND(1)*20)-10:VX=240:VY=175+UV:IF VY=175 THEN V=1
 340 SET [6] HX,HY:SET [6] VX,VY:U=0
 350 REM ―― MAIN ROUTIN ――
 360 IF B=0 GOTO 380
 370 GOSUB 1210:REM ―― UFO MOVE ――
 380 GET KY$:USR(1140):IF KY$="" GOSUB 770:GOTO 440
 390 IF KY$="2" GOSUB 470:GOTO 440
 400 IF KY$="4" GOSUB 510:GOTO 440
 410 IF KY$="6" GOSUB 540:GOTO 440
 420 IF KY$="8" GOSUB 590:GOTO 440
 430 IF KY$=" " GOSUB 650:GOTO 440
 440 IF U=1 GOTO 300:REM ―― UFO ﾊｯｾｲ ――
 450 GOTO 360
 460 REM ―― V ﾚｰﾀﾞｰ ｺｳｶ ――
 470 IF (H=1)*(V=1) GOSUB 990:RETURN
 480 FOR I=0 TO 500:NEXT:RESET [6,W1] VX,VY:VY=VY-1:IF VY<155 THEN U=1:H=0:RESET
 [6,W1] HX,HY:RETURN
 490 SET [6] VX,VY:GOSUB 620:RETURN
 500 REM ―― H ﾚｰﾀﾞｰ ﾋﾀﾞﾘ ｾﾝｶｲ ――
 510 IF (H=1)*(V=1) GOSUB 1030:RETURN
 520 FOR I=0 TO 500:NEXT:RESET [6,W1] HX,HY:A=A+10:GOSUB 1150:SET [6] HX,HY:GOSU
B 560:RETURN
 530 REM ―― H ﾚｰﾀﾞｰ ﾐｷﾞ  ｾﾝｶｲ ――
 540 IF (H=1)*(V=1) GOSUB 1070:RETURN
 550 FOR I=0 TO 500:NEXT:RESET [6,W1] HX,HY:A=A-10:GOSUB 1150:SET [6] HX,HY:GOSU
B 560:RETURN
 560 IF A=270 THEN H=1:SET [6] HX,HY:RETURN
 570 H=0:RETURN
 580 REM ―― V ﾚｰﾀﾞｰ ｼﾞｮｳｼｮｳ ――
 590 IF (H=1)*(V=1) GOSUB 1110:RETURN
 600 FOR I=0 TO 500:NEXT:RESET [6,W1] VX,VY:VY=VY+1:IF VY>195 THEN U=1:H=0:RESET
 [6,W1] HX,HY:RETURN
 610 SET [6] VX,VY:GOSUB 620:RETURN
 620 IF VY=175 THEN V=1:SET [6] VX,VY:RETURN
 630 V=0:RETURN
 640 REM ―― ﾋﾞｰﾑ ﾊｯｼｬ ――
 650 IF FU=1 THEN RETURN
 660 MUSIC"+D1":LINE [2] 0,150,328,84:LINE [2] 639,150,328,84:BLINE [2,W1] 0,150
,328,84:BLINE [2,W1] 639,150,328,84
 670 EN=EN+1:BLINE [6,W1] 105,154+EN,145,154+EN:IF EN>40 THEN FU=1
 680 IF (X>17)*(X<21)*(Y>7)*(Y<11) GOTO 700
 690 RETURN
 700 USR($F838):POKE $FFFF,1:USR($F800):IF (X=19)*(Y=9) THEN SC=SC+200:GOTO 720
 710 SC=SC+100
 720 MUSIC "C1DCDCD":USR($F838):U=1:B=0:H=0:V=0:X=0:Y=0
 730 IF SC>HI THEN HI=SC
 740 CURSOR  6,0:PRINT SC:CURSOR 34,0:PRINT HI
 750 RETURN
 760 REM ―― UFO ｺｳｹﾞｷ ――
 770 TI=VAL(TI$):IF TI<15 THEN RETURN
 780 IF TI>45 GOTO 850
 790 IF TI>30 GOTO 920
 800 IF (B=0)*(R=20) THEN RETURN
 810 IF B=0 THEN R=20:RESET [6,W1] HX,HY:GOSUB 1150:SET [6] HX,HY:GOTO 950
 820 GOTO 830
 830 C=INT(RND(1)*10):IF C<>0 THEN RETURN
 840 C=INT(RND(1)*640):MUSIC"+C":LINE [2] X*16+16,Y*8+8,C,149:BLINE [2,W1] X*16+
16,Y*8+8,C,149:IF (C<150)+(C>490) THEN RETURN
 850 GRAPHO1:MUSIC"C+C":GRAPHO2:MUSIC"C+C":GRAPHO3:MUSIC"C+C":COLOR,07
 860 IF B=0 THEN RESET [6,W1] HX,HY:RESET [6,W1] VX,VY
 865 IF B=1 THEN USR($F838)
 870 DM=DM+1:IF DM>2 THEN GOSUB 1500:REM ―― GAME OVER ――:RETURN
 880 IF DM>1 THEN BOX [2]494,155,534,195,F2:GOTO 910
```

```
890 IF DM>0 THEN BOX [6]494,155,534,195,F6
900 IF B=0 THEN RESET [2,W1] HX,HY
910 H=0:V=0:B=0:U=1:RETURN
920 IF (B=0)*(R=10) THEN RETURN
930 IF B=0 THEN R=10:RESET [6,W1] HX,HY:GOSUB 1150:SET [6] HX,HY:GOTO 950
940 GOTO 830
950 RESET [6,W1] VX,VY
960 IF VY>175 THEN VY=175+(VY-175)/2:SET [6] VX,VY:RETURN
970 VY=175-(175-VY)/2:SET [6] VX,VY:RETURN
980 REM ―― ｺｳｶ ――
990 IF B=0 THEN X=INT(RND(1)*38):Y=15:CURSOR X,Y:POKE $FFFF,0:USR($F800):B=1:RE
TURN
1000 CURSOR X,Y:USR($F838):Y=Y-1:IF Y<0 THEN V=0:B=0:RETURN
1010 CURSOR X,Y:USR($F800):RETURN
1020 REM ―― ﾋﾀﾞﾘ ｾﾝｶｲ ――
1030 IF B=0 THEN X=0:Y=INT(RND(1)*16):CURSOR X,Y:POKE $FFFF,0:USR($F800):B=1:RE
TURN
1040 CURSOR X,Y:USR($F838):X=X+1:IF X>37 THEN H=0:B=0:RETURN
1050 CURSOR X,Y:USR($F800):RETURN
1060 REM ―― ﾐｷﾞ  ｾﾝｶｲ ――
1070 IF B=0 THEN X=37:Y=INT(RND(1)*16):CURSOR X,Y:POKE $FFFF,0:USR($F800):B=1:R
ETURN
1080 CURSOR X,Y:USR($F838):X=X-1:IF X<0 THEN H=0:B=0:RETURN
1090 CURSOR X,Y:USR($F800):RETURN
1100 REM ―― ｼﾞｮｳｼｮｳ ――
1110 IF B=0 THEN X=INT(RND(1)*36):Y=0:CURSOR X,Y:POKE $FFFF,0:USR($F800):B=1:RE
TURN
1120 CURSOR X,Y:USR($F838):Y=Y+1:IF Y>15 THEN V=0:B=0:RETURN
1130 CURSOR X,Y:USR($F800):RETURN
1140 REM ―― H ﾚｰﾀﾞｰ ｻﾞﾋｮｳ ｹｲｻﾝ ――
1150 IF A=-10 THEN A=350
1160 IF A=360 THEN A=0
1170 UH=A*π/180
1180 HX=400+R*COS(UH)
1190 HY=175+R*SIN(UH)*.41
1200 RETURN
1210 REM ―― UFO MOVE ――
1220 XX=INT(RND(1)*3)-1
1230 YY=INT(RND(1)*3)-1
1240 CURSOR X,Y:USR($F838)
1250 X=X+XX:IF X<0 THEN X=0
1260 IF X>37 THEN X=37
1270 Y=Y+YY:IF Y<0 THEN Y=0
1280 IF Y>15 THEN Y=15
1290 CURSOR X,Y:USR($F800)
1300 RETURN
1310 REM ―― TITLE ――
1320 CCOLOR 6:RESTORE 1420
1330 READ CX,CY:IF (CX=99)*(CY=99) GOTO 1350
1340 CURSOR CX,CY:PRINT CHR$(30):GOTO 1330
1350 A$="              ｹﾞ ｷ ﾂ ｲ  ｹﾞ ｰ ﾑ          "
1360 FOR I=0 TO 37:CURSOR I,15:POKE $FFFF,0:USR($F800):B$=MID$(A$,I+1,1):USR($F
838):CURSOR I,16:PRINT B$:NEXT
1370 CURSOR 15,24:PRINT "HIT ANY KEY";
1380 GET A$:IF A$="" THEN CURSOR 15,24:PRINT "             ";:GOTO 1370
1390 FOR I=0 TO 10:FOR J=0 TO 39:CURSOR J,I:PRINT CHR$(32):NEXT J,I
1400 RETURN
1410 REM ―― U ――
1420 DATA 0,0,0,1,0,2,0,3,0,4,0,5,0,6,0,7,0,8,0,9,0,10,1,10,2,10,3,10,4,10,5,10
,6,10,7,10,8,10,9,10,10,10
1430 DATA 10,9,10,8,10,7,10,6,10,5,10,4,10,3,10,2,10,1,10,0
1440 REM ―― F ――
1450 DATA 15,10,15,9,15,8,15,7,15,6,15,5,15,4,15,3,15,2,15,1,15,0,16,0,17,0,18,
0,19,0,20,0,21,0,22,0,23,0,24,0,25,0
1460 DATA 16,5,17,5,18,5,19,5,20,5,21,5,22,5,23,5
1470 REM ―― O ――
1480 DATA 30,0,30,1,30,2,30,3,30,4,30,5,30,6,30,7,30,8,30,9,30,10,31,10,32,10,3
3,10,34,10,35,10,36,10,37,10,38,10,39,10
1490 DATA 39,9,39,8,39,7,39,6,39,5,39,4,39,3,39,2,39,1,39,0,38,0,37,0,36,0,35,0
,34,0,33,0,32,0,31,0,99,99
1500 REM ―― GAME OVER ――
1510 CURSOR 12,15:PRINT"―― GAME OVER ――":CURSOR 10,17:PRINT"ﾓｳｲﾁﾄﾞ ﾁｬﾚﾝｼﾞ ｼﾏｽ
ｶ (Y/N)"
1520 GET A$:IF A$="N" THEN GRAPH C7:PRINT CHR$(6):END
1530 IF A$<>"Y" GOTO 1520
1540 U=1:SC=0:EN=0:B=0:H=0:Y=0:DM=0:FU=0:BOX [6] 105,155,145,195,F6:BOX 494,155
,534,195,F5
1550 CURSOR 12,15:PRINT"                    ":CURSOR 10,17:PRINT"
        ":CURSOR 7,0:PRINT "0       "
1560 RETURN
1570 REM ―― ｹﾞｰﾑ ｾﾂﾒｲ ――
1580 CCOLOR 7:PRINT"●H⇩⇩"
```

```
1590 PRINT"  ┌───┬─┬─┐            ┌─────┐   ┌─┬─┬─┬─┐"
1600 PRINT"  |ｶﾅ| 1| 2|............|BREAK|   |7|■|9|+|"
1610 PRINT"  ├───┴┬┴┬┘            ├─┬─┬─┤   ├─┼─┼─┼─┤"
1620 PRINT"  |GRP| Q|.............|[|H|D|   |■|5|■|-|"
1630 PRINT"  ├────┴┬┴┐          ┌┴┬┴─┴─┤   ├─┼─┼─┼─┤"
1640 PRINT"  |SFLK| A|..........|]|  CR |   |1|■|3|E|"
1650 PRINT"  ├─────┴┬┴┐        ┌┴─┴────┤   ├─┼─┼─┤N|"
1660 PRINT"  |SHIFT| Z|..........|SHIFT|   | | | .|T|"
1670 PRINT"  └──────┴─┘          └─────┘   └─┴─┴─┴─┘"
1680 PRINT"      ┌─┬──────────────────┐"
1690 PRINT"      |T|    ■ ■ ■ ■ ■      |"
1700 PRINT"      └─┴──────────────────┘"
1710 PRINT:PRINT
1720 PRINT"               8  ...  ｼﾞｮｳｼｮｳ   "
1730 PRINT"               6  ...  ﾐｷﾞ   ｾﾝｶｲ"
1740 PRINT"               4  ...  ﾋﾀﾞﾘ ｾﾝｶｲ"
1750 PRINT"               2  ...  ｺｳｶ       "
1760 PRINT"           SPACE  ...  ﾋﾞｰﾑ      "
1770 CURSOR 15,24:PRINT"HIT ANY KEY";
1780 GET A$:IF A$="" THEN CURSOR 15,24:PRINT"             ";:GOTO 1770
1790 RETURN
```

リスト 2 マシン語DATA

```
** Oh!MZ Check sum system V1.0 **

ADR. +0 +1 +2 +3 +4 +5 +6 +7 :sum.
F370 00 00 00 00 00 00 00 00 :0.00
F378 00 00 00 19 06 00 00 00 :0.1F
F380 FF 00 00 00 FF 00 00 00 :1.FE
F388 FF 00 00 00 FF 00 00 00 :1.FE
F390 FF 00 00 00 FF 00 00 00 :1.FE
F398 FF 00 00 00 FF 00 00 00 :1.FE
F3A0 FF 00 00 00 FF 00 00 00 :1.FE
F3A8 FF 00 00 00 FF 00 00 00 :1.FE
-----------------------------------0C13
F3B0 FF 00 00 00 FF 00 00 00 :1.FE
F3B8 FF 00 00 00 FF 00 00 00 :1.FE
F3C0 FF 00 00 00 FF 00 00 00 :1.FE
F3C8 FF 00 00 00 FF 00 F0 00 :2.EE
F3D0 0F 00 0F 00 F0 00 00 00 :1.0E
F3D8 FF 00 00 00 FF 00 00 00 :1.FE
F3E0 FF 00 C0 40 3F 00 FF 55 :3.92
F3E8 00 00 FF 55 00 00 03 01 :1.58
-----------------------------------10DE
F3F0 FC 00 00 00 FF 00 00 00 :1.FB
F3F8 FF 00 FC 54 03 00 FF 55 :3.A6
F400 00 00 FF 55 00 00 3F 15 :1.A8
F408 C0 00 00 00 FF 00 00 00 :1.BF
F410 FF 00 FF 55 00 00 FF 55 :3.A7
F418 00 00 FF 55 00 00 FF 55 :2.A8
F420 00 00 00 00 FF 00 00 00 :0.FF
F428 FF 00 FC FC 03 00 FF FF :4.F8
-----------------------------------154E
F430 00 00 FF FF 00 00 3F 3F :2.7C
F438 C0 00 00 00 FF 00 00 00 :1.BF
F440 FF 00 3C 3C C3 00 3C 3C :2.B2
F448 C3 00 3C 3C C3 00 3C 3C :2.76
F450 C3 00 00 00 FF 00 00 00 :1.C2
F458 FF 00 3C 3C C3 00 3C 3C :2.B2
F460 C3 00 3C 3C C3 00 3C 3C :2.76
F468 C3 00 00 00 FF 00 00 00 :1.C2
-----------------------------------120F
F470 FF 00 FC FC 03 00 FF FF :4.F8
F478 00 00 FF FF 00 00 3F 3F :2.7C
F480 C0 00 00 00 FF 00 C0 40 :2.BF
F488 3F 00 FF 55 00 00 FF 55 :2.E7
F490 00 00 FF 55 00 00 FF 55 :2.A8
F498 00 00 03 01 FC 00 F0 50 :2.40
F4A0 0F 00 FF 55 00 00 FF 55 :2.B7
F4A8 00 00 FF 55 00 00 FF 55 :2.A8
-----------------------------------1761
F4B0 00 00 0F 05 F0 00 FC 54 :2.54
F4B8 03 00 FF 55 00 00 FF 55 :2.AB
F4C0 00 00 FF 55 00 00 FF 55 :2.A8
F4C8 00 00 3F 15 C0 00 FF 55 :2.68
F4D0 00 00 FF 55 00 00 FF 55 :2.A8
F4D8 00 00 FF 55 00 00 FF 55 :2.A8
F4E0 00 00 FF 55 00 00 FF 55 :2.A8
F4E8 00 00 FF 55 00 00 FF 55 :2.A8
-----------------------------------14AF
F4F0 00 00 FF 55 00 00 FF 55 :2.A8
F4F8 00 00 FF 55 00 00 FF 55 :2.A8
F500 00 00 FF 55 00 00 FF 55 :2.A8
F508 00 00 FF 55 00 00 FF 55 :2.A8
F510 00 00 FF 55 00 50 00 F0 :2.94
F518 0F 15 00 3F C0 54 00 FC :2.73
F520 03 05 00 0F F0 54 00 FC :2.57
F528 03 05 00 0F F0 40 00 C0 :2.07
-----------------------------------1405
F530 3F 05 00 0F F0 50 00 F0 :2.83
F538 0F 01 00 03 FC 50 00 F0 :2.4F
F540 0F 01 00 03 FC 00 00 00 :1.0F
F548 FF 00 00 00 FF 00 00 00 :1.FE
F550 FF 00 00 00 FF 00 00 00 :1.FE
F558 FF 00 00 00 FF 00 00 00 :1.FE
F560 FF 00 00 00 FF 00 00 00 :1.FE
F568 FF 00 00 00 FF 00 00 00 :1.FE
-----------------------------------0FD7
F570 FF 00 00 00 FF 00 00 00 :1.FE
F578 FF 00 00 00 FF 00 00 00 :1.FE
F580 FF 00 00 00 FF 00 00 00 :1.FE
F588 FF 00 00 00 FF 00 00 00 :1.FE
F590 FF 00 00 00 FF 00 00 00 :1.FE
F598 FF 00 00 00 FF 00 00 00 :1.FE
F5A0 FF 00 00 00 FF 00 00 00 :1.FE
F5A8 FF 00 00 00 FF 00 00 00 :1.FE
-----------------------------------0FF0
F5B0 FF 00 00 00 FF 00 00 00 :1.FE
F5B8 FF 00 00 00 FF 19 06 00 :2.1D
F5C0 00 00 FF 00 00 00 FF 00 :1.FE
F5C8 00 00 FF 00 0F 0F F0 00 :2.0D
F5D0 00 00 FF 00 C0 C0 3F 00 :2.BE
F5D8 0C 0C F3 00 00 00 FF 00 :2.0A
F5E0 00 00 FF 00 3F 3F C0 00 :2.3D
F5E8 00 00 FF 00 F0 F0 0F 00 :2.EE
-----------------------------------1219
F5F0 F0 F0 0F 00 00 00 FF 00 :2.EE
F5F8 C0 C0 3F 0C FF F3 00 00 :3.BD
F600 00 00 FF 00 3F 3F C0 00 :2.3D
```

```
F608 C0 C0 3F 00 0F 0F F0 00 :2.CD
F610 FC FC 03 0F FF F0 00 00 :3.F9
F618 FF FF 00 00 0F 0F F0 00 :3.0C
F620 C0 C0 3F 00 FF FF 00 F0 :4.AD
F628 FF 0F 00 3F FF C0 00 00 :3.0C
----------------------------------1A73
F630 FF FF 00 00 03 03 FC 00 :3.00
F638 00 00 FF 00 FF FF 00 FC :3.F9
F640 FF 03 00 FC FF 00 00 0F :3.0C
F648 FF F0 00 00 03 03 FC 00 :2.F1
F650 00 00 FF F0 FF 0F 00 FF :3.FC
F658 FF 00 00 FC FF 00 00 03 :2.FD
F660 FF FC 00 00 03 03 FC 00 :2.FD
F668 00 00 FF F0 FF 0F 00 0F :3.0C
----------------------------------19F8
F670 FF 00 00 F0 FF 00 00 00 :2.EE
F678 FF FF 00 00 00 00 FF 00 :2.FD
F680 00 00 FF C0 FF 3F 00 0F :3.0C
F688 FF 00 00 F0 FF 00 00 00 :2.EE
F690 FF FF 00 00 00 00 FF 00 :2.FD
F698 C0 C0 3F F0 FF 0F 00 03 :3.C0
F6A0 FF 00 00 F0 FF 00 00 03 :2.F1
F6A8 FF FC 00 00 03 03 FC 00 :2.FD
----------------------------------1890
F6B0 C0 C0 3F F0 FF 0F 00 03 :3.C0
F6B8 FF 00 00 C0 FF 00 00 3F :2.FD
F6C0 FF C0 00 00 0F 0F F0 00 :2.CD
F6C8 F0 F0 0F 3F FF 00 00 00 :3.2D
F6D0 FF 00 00 00 FF 00 00 FF :2.FD
F6D8 FF 00 00 00 3F 3F C0 C0 :2.FD
F6E0 FF 3F 00 0F FF 00 00 00 :2.4C
F6E8 FF 00 00 00 FF 00 00 F0 :2.EE
----------------------------------17EB
F6F0 FF 00 00 0F FF F0 00 00 :2.FD
F6F8 F0 F0 0F FF FF 00 00 00 :3.ED
F700 FF 00 00 00 FF 00 00 FC :2.FA
F708 FF 00 00 03 3F 3C C0 00 :2.3D
F710 C0 C0 3F F0 FF 0F 00 03 :3.C0
F718 FF 00 00 00 FF 00 00 FF :2.FD
F720 FF 00 00 00 0F 0F F0 00 :2.0D
F728 00 00 FF C0 FF 3F 00 03 :3.00
----------------------------------17EB
F730 FF 00 00 00 FF 00 00 3C :2.3A
F738 FF C0 00 00 0F 0F F0 00 :2.CD
F740 00 00 FF C0 FF 3F 00 33 :3.30
F748 FF 00 00 F0 FF 00 00 3F :3.2D
F750 FF C0 00 00 03 03 FC 00 :2.C1
F758 00 00 FF C0 FF 3F 00 FF :3.FC
F760 FF 00 00 FC FF 00 00 3F :3.39
F768 FF C0 00 00 03 03 FC 00 :2.C1
----------------------------------181B
F770 C0 C0 3F F0 FF 0F 00 C3 :4.80
F778 FF 3C 00 0F FF F0 00 30 :3.69
F780 FF CF 00 00 03 03 FC 00 :2.D0
F788 C0 C0 3F 3C FF C3 00 C0 :4.7D
F790 FF 3F 00 03 FF FC 00 00 :3.3C
F798 FF FF 00 00 0F 0F F0 00 :3.0C
F7A0 F0 F0 0F 00 FF FF 00 C0 :4.AD
F7A8 FF 3F 00 03 FF FC 00 00 :3.3C
----------------------------------1D67
F7B0 FF FF 00 00 0F 0F F0 00 :3.0C
F7B8 F0 F0 0F 00 03 03 FC 00 :2.F1
F7C0 F0 F0 0F 00 FF FF 00 00 :3.ED
F7C8 C0 C0 3F 00 3F 3F C0 00 :2.FD
F7D0 3C 3C C3 00 00 00 FF 00 :2.3A
F7D8 C0 C0 3F 00 0F 0F F0 00 :2.CD
F7E0 00 00 FF 00 F0 F0 0F 00 :2.EE
F7E8 0F 0F F0 00 00 00 FF 00 :2.0D
----------------------------------16E9
F7F0 C0 C0 3F 00 03 03 FC 00 :2.C1
F7F8 00 00 FF 00 C0 C0 3F 00 :2.BE
F800 CD 5E F8 EB 4E 23 46 23 :3.EB
F808 EB 0D C5 E5 EB C5 D5 56 :5.7D
F810 23 5E 23 D5 C1 56 23 5E :3.11
F818 23 E3 E5 D5 11 00 B0 ED :4.6E
F820 52 D1 E1 D4 40 11 D1 C1 :4.BB
F828 23 10 E1 E1 01 50 00 09 :2.4F
----------------------------------1D6D
F830 C1 79 3D B7 20 D3 C9 00 :3.EA
F838 CD 5E F8 EB 4E 23 46 EB :4.B0
F840 0D C5 E5 13 13 13 13 1A :2.1D
F848 D5 5F CD 45 F9 D1 23 10 :4.43
F850 F2 E1 01 50 00 09 C1 79 :3.67
F858 3D B7 20 E4 C9 00 3A FF :3.FA
F860 FF 87 16 00 5F 21 58 F9 :3.6D
F868 19 E5 ED 4B D1 11 21 80 :3.B9
----------------------------------1D81
F870 BD 11 80 02 04 19 10 FD :2.7A
F878 09 09 18 5F 00 DB E8 CB :3.17
F880 FF CB B7 D3 E8 3E 01 D3 :5.4E
F888 F7 7E A3 B0 77 3E 02 D3 :4.52
F890 F7 7E A3 B1 77 3E 03 D3 :4.54
F898 F7 7E A3 B2 77 DB E8 CB :5.CF
F8A0 BF CB F7 D3 E8 C9 DB E8 :6.C8
F8A8 CB FF CB B7 D3 E8 3E 01 :5.46
----------------------------------2562
F8B0 D3 F7 7E A3 77 3E 02 D3 :4.75
F8B8 F7 7E A3 77 3E 03 D3 F7 :4.9A
F8C0 7E A3 77 DB E8 CB BF CB :5.B0
F8C8 F7 D3 E8 C9 00 21 7D F8 :5.11
F8D0 11 40 11 01 50 00 ED B0 :2.50
F8D8 C9 B1 00 D1 E5 EB 5E 23 :4.9C
F8E0 56 00 E1 C9 00 2A 20 50 :2.9A
F8E8 41 54 54 45 52 4E 20 57 :2.45
----------------------------------1F9B
F8F0 52 49 54 45 52 20 20 56 :2.1C
F8F8 32 2E 30 20 20 42 79 20 :1.AB
F900 50 75 63 6B 20 4D 61 6E :2.CF
F908 2E 2F 4C 4F 56 45 20 4E :2.01
F910 41 4F 4D 49 20 4B 41 57 :2.29
F918 41 53 48 49 4D 41 2F 54 :2.36
F920 41 46 CF B2 BA DD B8 D7 :5.2E
F928 CC DE 20 B6 B8 C1 AE B3 :5.5A
----------------------------------177E
F930 20 B5 7B C3 DE C4 B3 21 :4.89
F938 21 21 2F 31 39 38 33 2E :1.74
F940 32 2E 31 32 00 E5 D5 11 :2.8E
F948 00 B0 ED 52 D1 E1 D4 69 :4.DE
F950 11 C9 CD CD F8 C3 B1 00 :4.E0
F958 7B F3 BD F5 FF F7 FF 00 :6.15
F960 00 00 00 00 00 00 00 00 :0.00
F968 00 00 00 00 00 00 00 00 :0.00
----------------------------------185E
F970 00 00 00 00 00 00 00 00 :0.00
F978 00 00 00 00 00 00 00 00 :0.00
F980 00 00 00 00 00 00 00 00 :0.00
F988 00 00 00 00 00 00 00 00 :0.00
F990 00 00 00 00 00 00 00 00 :0.00
F998 00 00 00 00 00 00 00 00 :0.00
F9A0 00 00 00 00 00 00 00 00 :0.00
F9A8 00 00 00 00 00 00 00 00 :0.00
----------------------------------0000
F9B0 00 00 00 00 00 00 00 00 :0.00
F9B8 00 00 00 00 00 00 00 00 :0.00
F9C0 00 00 00 00 00 00 00 00 :0.00
F9C8 00 00 00 00 00 00 00 00 :0.00
F9D0 00 00 00 00 00 00 00 00 :0.00
F9D8 00 00 00 00 00 00 00 00 :0.00
F9E0 00 00 00 00 00 00 00 00 :0.00
F9E8 00 00 00 00 00 00 00 00 :0.00
----------------------------------0000
F9F0 00 00 00 00 00 00 00 00 :0.00
F9F8 00 00 00 00 00 00 00 00 :0.00
FA00 00 00 00 00 00 00 00 00 :0.00
FA08 00 00 00 00 00 00 00 00 :0.00
FA10 00 00 00 00 00 00 00 00 :0.00
FA18 00 00 00 00 00 00 00 00 :0.00
FA20 00 00 00 00 00 00 00 00 :0.00
FA28 00 00 00 00 00 00 00 00 :0.00
----------------------------------0000
```

Dr. Kuの ハード入門 MZ-80K/C/1200/700

パラレルI/Oボードの製作

早稲田大学航空宇宙研究会 桒野雅彦

メモリボード補足

CPUボードに引き続き，メモリボードも完成しましたね。

9月号で書き忘れたのですが，メモリボードの6116は，F000HからFFFFHの4Kバイトに配置されます。このエリアは，MZ-80K/C，1200では，フロッピーディスクを使わない限り空きエリアになっていますので，6116だけ差し込むことで，メモリ容量を増やすことができます。また，9月号で説明したように，2732は2716とピン配置が酷似しています。6116も，2716とピン配置がほとんど同じであり，6116用にメモリボードを作成すると，2716はそのまま差し込めるほどです。一方，2732は2716よりアドレスピンが1本多いので，6116用のソケットにそのまま差し込むと，メモリ容量の半分しか使うことができませんが，一応使用可能です。

6116と2732のピン配置の上での決定的な違いは21番ピンで，2732ではアドレスピンA11であるのに対し，6116では書き込み信号用の$\overline{WE}$となっています。

したがって，6116用のソケットに2732を差し込むと，読み出し時は$\overline{WE}=H$となりますので，2732の前半2Kバイトが読み出されます。

9月号で作成したメモリボードの場合は2732の21番ピンを，カードエッジのA11端子（B−8）と接続して，6116のソケットに差し込んだ場合，F000Hからの4KバイトをROMにすることができます。F000Hに00Hを書き込んでおけば，モニタからFD（MZ-700ではF）コマンドによってF001H以後に書かれたプログラムを実行させることができます。

4Kバイトといえば，簡単なアセンブラ，逆アセンブラ，整数型のインタプリタなどは充分に入ってしまう大きさです。ROM中にそれらを書き込んでおいて，FDコマンドで，それらがRAM上にブロック転送されて実行されるようにすれば，わずらわしいカセットからのロードから解放されます。

MZ-700の場合，このエリアに裏RAMが存在するため，裏RAMを使用するBASICなどを起動する場合はROMのデータと裏RAMのデータが重複することになり,うまくありません。この場合，メモリボードは抜いておいてください。

さて，メモリボードの補足が終わったところで，今月はパラレルI/Oボードを作ることにしましょう。

●仕様

パラレルI/Oポート用のICとしては，ROMライタの製作にも利用した8255を利用することにします。さしあたり，8255は1個（8ビット×3ポート）あれば充分なのですが，それだけではおもしろくないので，GI社のAY-3-8910というPSG（プログラマブル・サウンド・ジェネレータ）を追加してみました。

AY-3-8910はPSGとしてはもっともポピュラーで，使用例も多い上，I/Oポートを8ビット×2ポート持っており，またサウンド出力端子を使って，簡単なD-Aコンバータにすることもできる便利なICですから，使ってみるとなかなかおもしろいものです。

音質は，方形波がベースになっているだけに，お世辞にも良いとはいえませんが，ゲームの効果音作りや，とにかく簡単に自動演奏をさせてみたい場合などにはこれでも充分でしょう。

パラレルI/Oボードの仕様は図1のようになります。

図1　パラレルI/Oボード仕様

主要IC	8255(PPI)×2 AY-3-8910(PSG)×1
ポート数	デジタル8ビット×6 サウンド／アナログ出力×3
サウンド出力	MAX 0.5W 3和音+1ノイズ +1エンベロープ

●設計

ここで例によって，これらのポートを何番地に配置するかということが問題になります。

私の作ったI/Oボードは，MZ-700がまだ影も形もなかったころのもので，I/O空間(BASICやアセンブラのOUT命令で使われるアドレス空間)のF8HからFFHまでを使っていたのですが，この領域は，MZ-700では内部のカスタムICがプリンタ用のポートとして確保しているので使うわけにはいかないのです。そこで，急きょアドレスバスの第7ビットA_7にインバータを挿入しました。

この結果，アドレスの配置は78H～7FHとなります。PSGは，8255のBポートとCポートの下位2ビットを使ってコントロールすることにします。

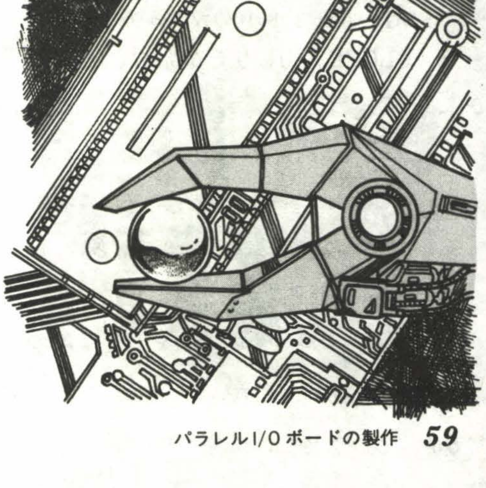
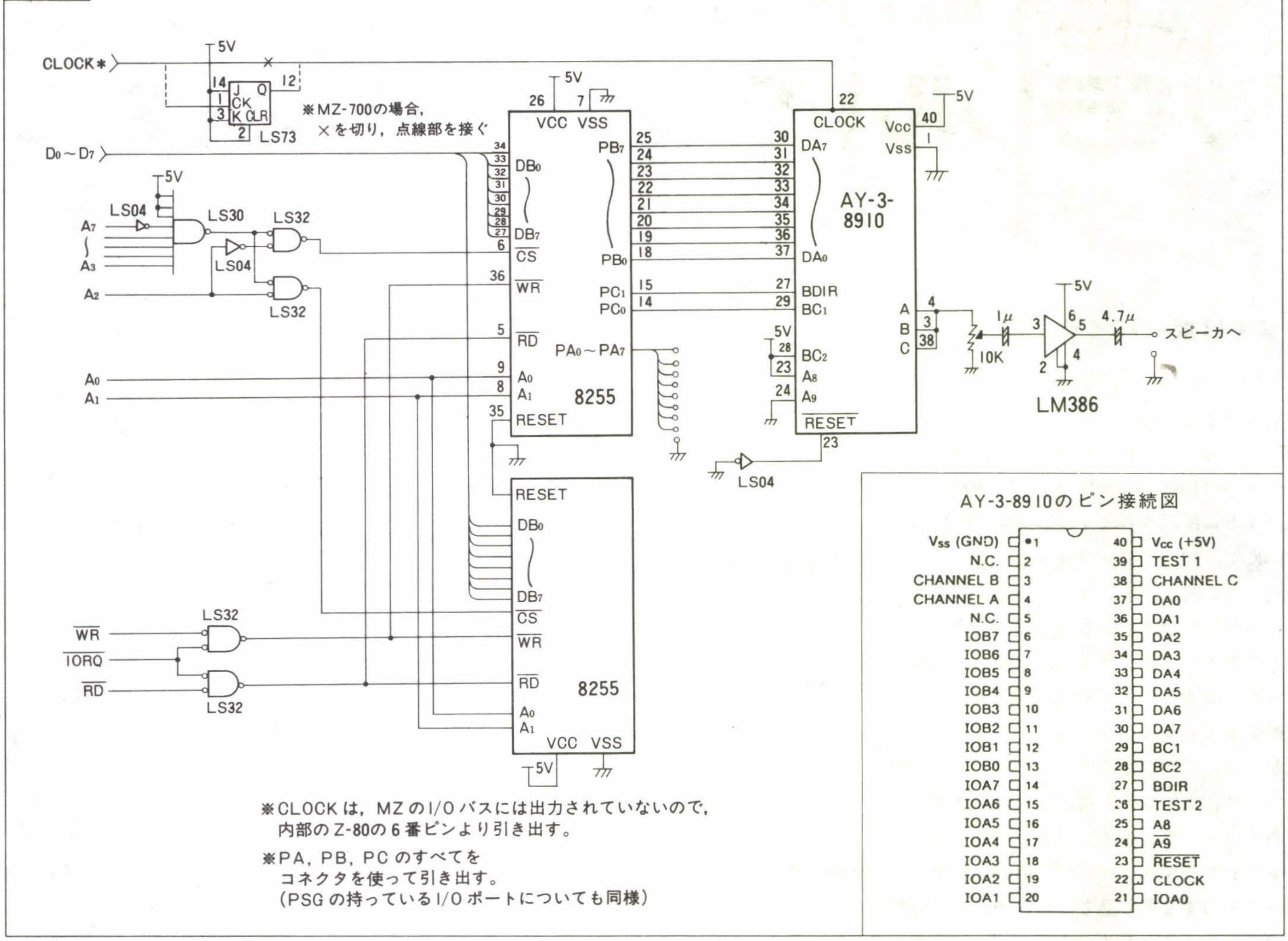


図2　**パラレルI/Oボード回路図**

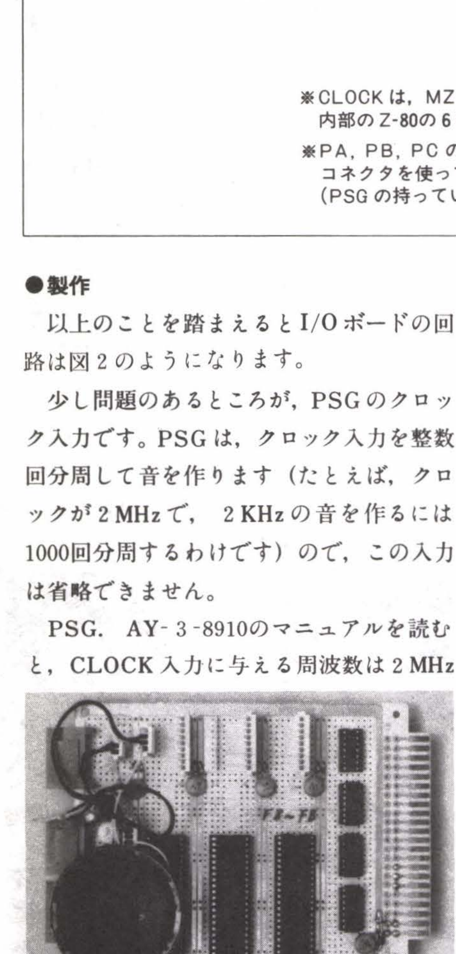


●製作

以上のことを踏まえるとI/Oボードの回路は図2のようになります。

少し問題のあるところが，PSGのクロック入力です。PSGは，クロック入力を整数回分周して音を作ります（たとえば，クロックが2MHzで，2KHzの音を作るには1000回分周するわけです）ので，この入力は省略できません。

PSG．AY-3-8910のマニュアルを読むと，CLOCK入力に与える周波数は2MHzまでとなっています。MZ-80K/C/1200の場合は，CPUに与えているクロックが2MHzですので，CPUのクロックをもらってくることで解決しますが，MZ-700の場合にはそういうわけにはいきません。MZ-700のCPUは3.579545MHz程度で動作していますので，これをそのまま使ったのではPSGの規格外となり，正常な動作は保障されません。



パラレルI/Oボード

対応策として，MZ-700の場合には，回路図のように74LS74を挿入し，周波数を$\frac{1}{2}$にすることにします。この場合，PSGに与える周波数は約1.79MHzとなり，MZ-80K／C／1200と同じデータを与えると音が低くなります。

各音階の周波数比は変化しませんので，一応，曲としては正しく演奏されますが，どうしても気になる人は，以前製作したCPUボードの回路図を参考に，2MHzのクロックを作成するとよいでしょう。

クロックについては，もうひとつ問題があります。MZの場合，背後の拡張バスにクロックがでていないのです（何ということだ)。仕方がないので，Z-80のクロック端子は直接ハンダづけをして引き出すことにします。Z-80のクロック端子は6番ピンです。隣のピンと仲良くさせたりしないように注意しながら，しっかりとハンダづけをしてクロックを引き出し，カードエッジ・コネクタのφ（A面2番）端子と接続します。

MZの中をさわるのがいやな人は，先ほど述べたようにCPUボードを参考にクロックを基板上で作成するとよいでしょう。

続いて，8255の$\overline{CS}$，$\overline{WR}$，$\overline{RD}$に与える信号作成部について説明しておきましょう。

$\overline{CS}$はChip-Select，$\overline{WR}$はWrite，$\overline{RD}$はReadの略で，$\overline{CS}$=Lで$\overline{WR}$=Lならば，A_0とA_1で選択されるポートにデータバスの値が書き込まれ，$\overline{CS}$=Lで$\overline{RD}$=Lな

ら，A_0とA$_1$で選択されるポートからデータを読み出します。当然，書き込むポートは出力に，読み出すポートは入力に設定しておかないと，データは無効になります。

$\overline{CS}$＝Hであったり，$\overline{CS}$＝Lでも$\overline{RD}$と$\overline{WR}$の両方がHである場合には8255のデータバスは，ハイ・インピーダンスのままです。ここでは$\overline{CS}$には，A_7～A_2のデコード出力を，$\overline{RD}$，$\overline{WR}$にはそれぞれ，CPUの$\overline{RD}$，$\overline{WR}$に，$\overline{IORQ}$（I/O Request）を加えた（両方ともLのときにLになるような）ものを与えています。これで，なぜI/O空間の78～7Fがポートになるかについては各自考えてみてください（ヒント：それぞれの8255の$\overline{CS}$，$\overline{RD}$，$\overline{WR}$がLになる条件を考える）。

各8255のポートを引き出すために，私は写真のような9ピンのコネクタを使い，ポートを8ビットと，GNDを引き出しましたが，10ピンのコネクタを使って，＋5Vもいっしょに引き出しておいたほうが良いかもしれません。

コネクタは秋葉原では，ネジ屋で買うことができます。いろいろなタイプがありますので，好みに応じて選ぶと良いでしょう。

●FINAL I

最近「実体配線図をつけてくれ」という投書を少なからず受け取っていますが，デジタル回路の場合，つながるべきところが確実につながっていれば，まず大抵動かないということはありません。ですから，部品配置は配線しやすいように設定すれば良いのです。

また，デジタル用，特に80系（8080，Z-80など）用のICは，ピン配列があまりきれいでないので，配線図ではすっきりしても実際の配線はかなりこみいったものになりがちです。

今のところこの連載にでてくる部品の種類はそれほど多くありませんし，回路図の読み方は，連載の最初にかなり詳しく説明しています。もう1度バックナンバーを読み返して，回路図に慣れておくことをおすすめします。

他の雑誌の製作記事などでもそうですが，回路を組む人は，回路図でものを考え，設計は回路図で行なわれ，実体配線図は製作したものから写していく場合がほとんどですから，どうしてもこみいったところではミスが発生しがちで，現実には，部品配置の参考くらいにしかならない場合がほとんどです。

●FINAL II

I/Oボードも完成したので，次回は少し息抜きをかねて，メカカメでも作ってみることにしましょう。

たぶん，この本がでるころは修学旅行シーズンで，東京に来られる方も多いかと思いますが，もし東京タワーに行くことがあれば，富士通のショールーム（3Fだったかなァ）を見にいってください。多分，手をたたくと集まってくるロボットが置いてあると思います。

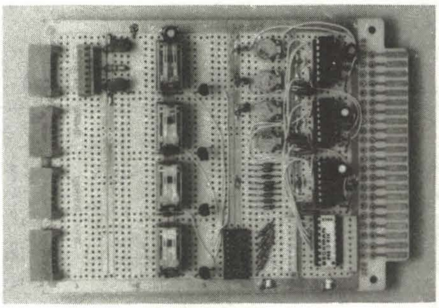
それと同じように，手をたたくとその方向を検出して寄ってくるロボット（？）を作ってみることにします。

現在までに完成した，インターフェースカード，また，パターンを起こしたI/Oボードの写真をのせておくことにしましょう。

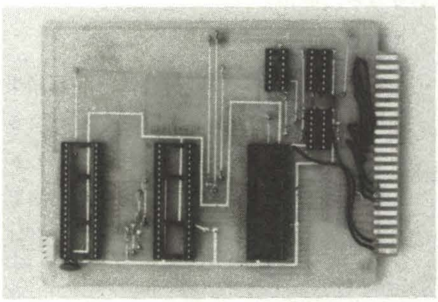
それではまた来月お目にかかりましょう。

部品表

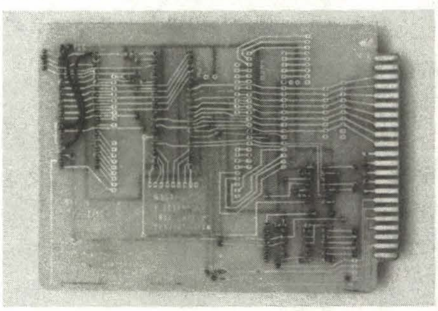
部品名	品番	メーカー	個数	単価(円)	入手先
I.C.	8255	――	2	300	亜土電子
	AY-3-8910	G・I	1	1200	〃
	74LS04	――	1	40	〃
	74LS30	――	1	60	〃
	74LS32	――	1	40	〃
	LM386	NS	1	150	〃
ソケット	40P	バンディ他	3	100	〃
	14P	〃	3	50	〃
コネクタ	9P	JST他	6	ペア50	ネジ屋(水谷等)
抵抗	1kΩ	――	1	10	亜土電子
半固定抵抗		――	1	50	〃
コンデンサ	4.7μF	――	2	30	〃
	0.1μF	――	5	15	〃
スピーカー	――	――	1	200	
基板	No.801-R	亜土電子	1	900	亜土電子



インターフェースカード



I/Oボードのパターンをおこしたもの(表)



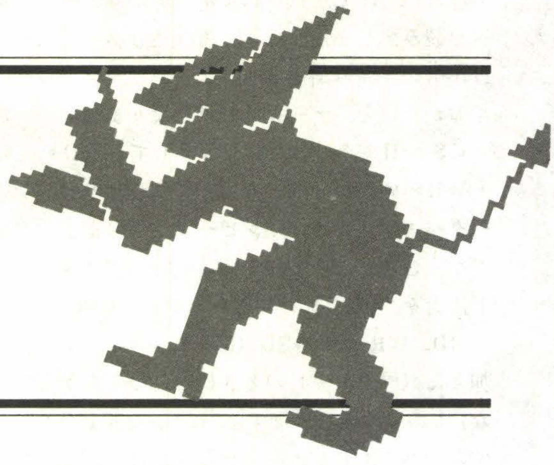
I/Oボードのパターンをおこしたもの(裏)

マシン語・魔神語

DEUS EX MACHINA 16ビット演算命令

連載第14回

長瀬 敏之



基礎編 「16ビット演算命令」

名前のとおり，16ビットの演算を行なう命令です。8ビット演算命令では，00H～FFHの範囲の数値しか扱えませんでしたが，この命令を使うことにより，0000H～FFFFHの範囲の数値を扱うことができます。

16ビット演算命令は，(LIST) のとおりです。

また，これらの命令の実行例は，(例1) のとおりです。

8ビット演算命令との違い

・(LIST)の注釈を見ればわかると思いますが，「ADD」・「ADC」・「SBC」・「INC」・「DEC」とも，8ビットが16ビットになっただけで，演算内容は，8ビット演算命令と同じです。

・フラグの変化

ADD命令

8ビット演算命令の場合は，演算結果がゼロ（A=00H）になるとゼロフラグはセット（Z=1）されます。しかし，16ビット演算命令の場合は，演算結果がどうであろうと，ゼロフラグは変化しません。

INC・DEC命令

16ビット演算命令の場合，どのフラグもまったく変化しません。

・どうでもいいことかもしれませんが，

```
    ADD  HL, BC
```

を，8ビット演算命令を使用して表現すると，次のようになります。

```
    LD   A, L
    ADD  A, C
    LD   L, A
    LD   A, H
    ADC  A, B
    LD   H, A
```

ただし，この場合レジスタAの内容が壊れます。

32ビットの演算方法

8ビットの演算命令で16ビットの演算ができるように，16ビットの演算命令を使用すれば，同じような方法で32ビットの演算ができます(PROGRAM 1)。

このプログラムは，(8000H～8003H)と(8004H～8007H) に書き込まれている2つの数値に対して，加算・減算を行ないます。そして，演算結果はそれぞれ（8008H～800BH）と（800CH～800FH）に書き込まれます。ただし，演算結果が00000000H～FFFFFFFFHの範囲の外に出る場合は，正解が得られません。ちなみに，FFFFFFFFHを10進数で表わすと，4294967295になります。

・例

17320508H＋14142135H＝？Ⓐ

17320508H－14142135H＝？Ⓑ

を求める場合，

まず，メモリに次のように書き込みます。

(8000H) ＝08H　(8004H) ＝35H

(8001H) ＝05H　(8005H) ＝21H

(8002H) ＝32H　(8006H) ＝14H

(8003H) ＝17H　(8007H) ＝14H

そして，「PROGRAM 1」を実行すると，8008H番地以降に演算結果が書き込まれます。

(8008H) ＝3DH　(800CH) ＝D3H

(8009H) ＝26H　(800DH) ＝E3H

(800BH)＝46H　(800EH)＝1DH

(800BH)＝2BH　(800FH)＝03H

したがって，求める答は次のようになります。

？Ⓐ＝2B46263DH

？Ⓑ＝031DE3D3H

16ビット演算命令を使って乗算を行なう

16ビット演算命令とロード命令を組み合わせることによって乗算を行なうことができます。その例をPROGRAM 2で示します。

注：各プログラムの前に，「HL＝HL＊n」と書いてありますが，これは「HLの内容をn倍にする」という意味です。

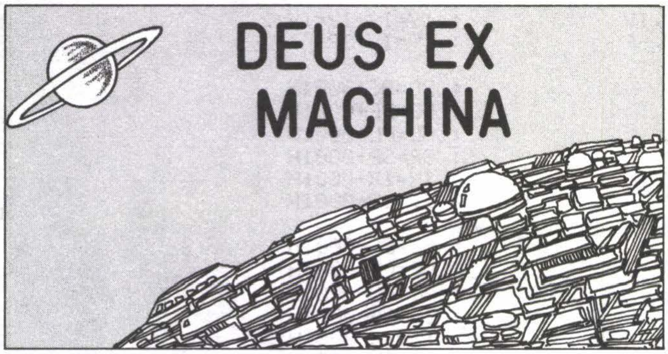
・演算結果がFFFFHを越えると，正しい答は得られません。

・「HL＝HL＊7」ではSBC命令を使っていますが，前のADD命令により，キャリフラグがリセット（CY＝0）されるので，結局「HL＝HL－DE－CY＝HL－DEになります。

このプログラムは，任意の数に対して乗算が行なえるような，まともな乗算プログラムではありません。ただ単に，加減算を組み合わせて乗算を行なっているだけです。しかし，機械語プログラム中で，わざわざまともな乗算プログラムを使用するほどのことでもないときは，これらの簡単なプログラムが役に立つと思います。

あとがき

次回は，基礎編をお休みにして，実践編「MZ-700特集」を予定しています。



例 1

ADD HL, BC

実行前 → 実行後

	実行前				実行後		
レジスタ H	59	B	3A	レジスタ H	9 4	B	3 A
L	63	C	BC	L	I F	C	B C

ADC HL, BC

実行前 → 実行後

	実行前				実行後		
レジスタ H	59	B	3A	レジスタ H	94	B	3A
L	63	C	BC	L	20	C	BC

フラグ　CY= I　　　フラグ　CY=ϕ

SBC HL, BC

実行前 → 実行後

	実行前				実行後		
レジスタ H	59	B	3A	レジスタ H	IE	B	3A
L	63	C	BC	L	A6	C	BC

フラグ　CY= I　　　フラグ　CY=ϕ

INC DE

実行前 → 実行後

	実行前		実行後
レジスタ D	59	レジスタ D	59
E	63	E	64

DEC IX

実行前 → 実行後

レジスタ IX 5963　　　レジスタ IX 5962

リスト

```
          **  Z80 ASSEMBLER SP-2102 PAGE 01  **

 01 0000                ;
 02 0000                ; LIST
 03 0000                ;
 04 0000                ;     16 ビット エンザン メイレイ
 05 0000                ;
 06 0000                ;
 07 0000 09                   ADD   HL,BC         ; HL=HL+BC
 08 0001 19                   ADD   HL,DE         ; HL=HL+DE
 09 0002 29                   ADD   HL,HL         ; HL=HL+HL=HL*2
 10 0003 39                   ADD   HL,SP         ; HL=HL+SP
 11 0004                ;
 12 0004 ED4A                 ADC   HL,BC         ; HL=HL+BC+CY
 13 0006 ED5A                 ADC   HL,DE         ; HL=HL+DE+CY
 14 0008 ED6A                 ADC   HL,HL         ; HL=HL+HL+CY=HL*2+CY
 15 000A ED7A                 ADC   HL,SP         ; HL=HL+SP+CY
 16 000C                ;
 17 000C ED42                 SBC   HL,BC         ; HL=HL-BC-CY
 18 000E ED52                 SBC   HL,DE         ; HL=HL-DE-CY
 19 0010 ED62                 SBC   HL,HL         ; HL=HL-HL-CY=-CY
 20 0012 ED72                 SBC   HL,SP         ; HL=HL-SP-CY
 21 0014                ;
 22 0014 DD09                 ADD   IX,BC         ; IX=IX+BC
 23 0016 DD19                 ADD   IX,DE         ; IX=IX+DE
 24 0018 DD29                 ADD   IX,IX         ; IX=IX+IX=IX*2
 25 001A DD39                 ADD   IX,SP         ; IX=IX+SP
 26 001C                ;
 27 001C FD09                 ADD   IY,BC         ; IY=IY+BC
 28 001E FD19                 ADD   IY,DE         ; IY=IY+DE
```

```
0020 FD29                ADD   IY,IY          ; IY=IY+IY=IY*2
0022 FD39                ADD   IY,SP          ; IY=IY+SP
0024                ;
0024 03                  INC   BC             ; BC=BC+0001H
0025 13                  INC   DE             ; DE=DE+0001H
0026 23                  INC   HL             ; HL=HL+0001H
0027 33                  INC   SP             ; SP=SP+0001H
0028 DD23                INC   IX             ; IX=IX+0001H
002A FD23                INC   IY             ; IY=IY+0001H
002C                ;
002C 0B                  DEC   BC             ; BC=BC-0001H
002D 1B                  DEC   DE             ; DE=DE-0001H
002E 2B                  DEC   HL             ; HL=HL-0001H
002F 3B                  DEC   SP             ; SP=SP-0001H
0030 DD2B                DEC   IX             ; IX=IX-0001H
0032 FD2B                DEC   IY             ; IY=IY-0001H
0034                ;
0034                     END
```

```
   **  Z80 ASSEMBLER SP-2102 PAGE 01  **

0000                ;
0000                ; PROGRAM 1
0000                ;
0000                ;    32BIT ｴﾝｻﾞﾝ
0000                ;
0000                ;       (8008H-800BH)←(8000H-8003H)+(8004H-8007H)
0000                ;       (800CH-800FH)←(8000H-8003H)-(8004H-8007H)
0000                ;
0000                     REL   7000H
7000                ;
7000                ;
7000 ED4B0480            LD    BC,(8004H)     ; B=(8005H),C=(8004H)
7004 ED5B0680            LD    DE,(8006H)     ; D=(8007H),E=(8006H)
7008                ;
7008 2A0080              LD    HL,(8000H)     ; H=(8001H),L=(8000H)
700B 09                  ADD   HL,BC          ; HL=HL+BC
700C 220880              LD    (8008H),HL     ; (8009H)=H,(8008H)=L
700F 2A0280              LD    HL,(8002H)     ; H=(8003H),L=(8002H)
7012 ED5A                ADC   HL,DE          ; HL=HL+DE+CY
7014 220A80              LD    (800AH),HL     ; (800BH)=H,(800AH)=L
7017                ;
7017 A7                  AND   A              ; CY=0
7018 2A0080              LD    HL,(8000H)     ; H=(8001H),L=(8000H)
701B ED42                SBC   HL,BC          ; HL=HL-BC-CY=HL-BC
701D 220C80              LD    (800CH),HL     ; (800DH)=H,(800CH)=L
7020 2A0280              LD    HL,(8002H)     ; H=(8003H),L=(8002H)
7023 ED52                SBC   HL,DE          ; HL=HL-DE-CY
7025 220E80              LD    (800EH),HL     ; (800FH)=H,(800EH)=L
7028                ;
7028 C30000              JP    0000H          ; GOTO 0000H
702B                ;
702B                ;
702B                     END
```

```
   **  Z80 ASSEMBLER SP-2102 PAGE 01  **

0000                ;
0000                ; PROGRAM 2
0000                ;
0000                ;
0000                ;
0000                ;  HL=HL*2
0000                ;
0000 29                  ADD   HL,HL          ; HL=HL+HL
0001                ;
0001                ;
0001                ;  HL=HL*3
```

```
0001                 ;
0001 54                       LD    D,H            ; DE=HL
0002 5D                       LD    E,L            ;
0003 19                       ADD   HL,DE          ; HL=HL+DE
0004 19                       ADD   HL,DE          ; HL=HL+DE
0005                 ;
0005                 ;
0005                 ;   HL=HL*4
0005                 ;
0005 29                       ADD   HL,HL          ; HL=HL+HL
0006 29                       ADD   HL,HL          ; HL=HL+HL
0007                 ;
0007                 ;
0007                 ;   HL=HL*5
0007                 ;
0007 54                       LD    D,H            ; DE=HL
0008 5D                       LD    E,L            ;
0009 29                       ADD   HL,HL          ; HL=HL+HL
000A 29                       ADD   HL,HL          ; HL=HL+HL
000B 19                       ADD   HL,DE          ; HL=HL+DE
000C                 ;
000C                 ;
000C                 ;   HL=HL*6
000C                 ;
000C 54                       LD    D,H            ; DE=HL
000D 5D                       LD    E,L            ;
000E 19                       ADD   HL,DE          ; HL=HL+DE
000F 19                       ADD   HL,DE          ; HL=HL+DE
0010 29                       ADD   HL,HL          ; HL=HL+HL
0011                 ;
0011                 ;
0011                 ;   HL=HL*7
0011                 ;
0011 54                       LD    D,H            ; DE=HL
0012 5D                       LD    E,L            ;
0013 29                       ADD   HL,HL          ; HL=HL+HL
0014 29                       ADD   HL,HL          ; HL=HL+HL
0015 29                       ADD   HL,HL          ; HL=HL+HL
0016 ED52                     SBC   HL,DE          ; HL=HL-DE-CY
```

```
   **  Z80 ASSEMBLER SP-2102 PAGE 02  **

0018                 ;
0018                 ;
0018                 ;   HL=HL*8
0018                 ;
0018 29                       ADD   HL,HL          ; HL=HL+HL
0019 29                       ADD   HL,HL          ; HL=HL+HL
001A 29                       ADD   HL,HL          ; HL=HL+HL
001B                 ;
001B                 ;
001B                 ;   HL=HL*9
001B                 ;
001B 54                       LD    D,H            ; DE=HL
001C 5D                       LD    E,L            ;
001D 29                       ADD   HL,HL          ; HL=HL+HL
001E 29                       ADD   HL,HL          ; HL=HL+HL
001F 29                       ADD   HL,HL          ; HL=HL+HL
0020 19                       ADD   HL,DE          ; HL=HL+DE
0021                 ;
0021                 ;
0021                 ;   HL=HL*10
0021                 ;
0021 54                       LD    D,H            ; DE=HL
0022 5D                       LD    E,L            ;
0023 29                       ADD   HL,HL          ; HL=HL+HL
0024 29                       ADD   HL,HL          ; HL=HL+HL
0025 19                       ADD   HL,DE          ; HL=HL+DE
0026 29                       ADD   HL,HL          ; HL=HL+HL
0027                 ;
0027                 ;
0027                          END
```

MZ-LOGO あれ・これ

数の取り扱いから音楽演奏まで

こうもと　やすひこ

1 はじめに

先月号と先々月号で，LOGOの大きな柱であるタートルグラフィックと記号処理に関して説明を行ないました。そこで，今月はそれらの落ち穂ひろい的な意味で，MZ-LOGOを実際に動かしてみて気がついた点を，書き並べてみたいと思います。MZ-LOGOのマニュアルは，よくできてはいますがかなり急いで作られたのか，少し不親切なところがあります。その点を少しでも埋めることができれば幸いです。

2 数の取り扱い

LOGOでは数値演算ができますが，数の入力には少し注意が必要です。

ーは引き算を表わす演算子ですが，それが

```
-2        -:X
```

などのように使われるときは，符号を逆転する演算子とみなすことができます。MZ-LOGOでは，このような演算は数値に対してしか認められていません。また，変数の前に－をつけることはできません。上の例でいえば，**-2**は許されるけれど，**-:x**は許されないことになります。

数値に－を付けて使う場合でも，それを(　)で囲まなければならないという制約があります。これは次のような例を考えれば明らかでしょう。

```
at 20 -10
```

はタートルの位置を（20，－10）の座標へ移す命令です。しかし，これではLOGOはまず20－10を計算してしまいます。なぜなら演算子（＋，－，＊，／）の前後にある空白には意味がないからです。それで，タートルを移すべきx座標は10（＝20－10）ですが，y座標がないためエラーとなってしまうのです。逆に

```
at -10 20
```

とした場合は，10を引くべき相手がないので，これもエラーとなります。ですから，引数間の区切りをはっきりさせるため(　)をつけて，

```
at 20 (-10)
at (-10) 20
```

としなければならないのです。

```
.123
```

と書いた場合，ふつうこれは0.１２３を意味します。しかし、MZ-LOGOでは小数点で始まる数は数とみなしてくれません。したがって，上の例は面倒でも

```
0.123
```

と入力しなければならないのです。0を省いたばかりにエラーとなってしまったなんてことはないようにしましょう。

次は，マニュアルでは触れられていない指数型の実数の入力について説明します。

MZ-LOGOは実数を指数の形で入力することもできます。そのときの形式は

数字e指数
数字E指数
数字n指数
数字N指数

の4種類です。意味は，eかEを使った場合が

数字$\times 10^{EN}$（EN：指数）

nかNを使った場合が

数字$\times 10^{-EN}$（EN：指数）

です。数字の部分は整数か普通の（指数型でない）実数です。指数の部分は負でない整数です。また，指数の範囲は，MZ-LOGOは16bitで数値計算を行なうので，0から38までとなっています。これが39以上になると数字の部分だけが意味を持つようです。

```
print 1e37
1.00000E37
print 1e39
1.
print 1e39+1
2.
```

3 エディタとプログラム

LOGOのプログラムは

```
to 手続き名
```

で書き始めることができます。このとき，コマンドとコマンドの間の空白は1個以上なら何個でもよいのです。このことで迷っている人がいるかも知れないので書いておきます。つまり，プログラム中で

```
if :n=o [stop]
```

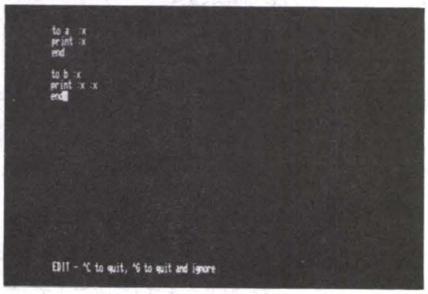
と書こうが，

```
if  :n =o  [stop]
```

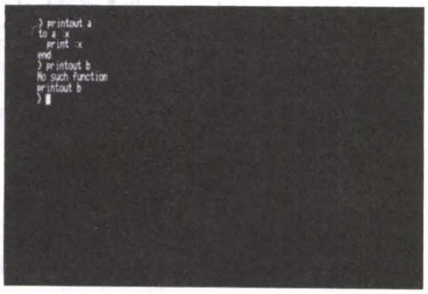
と書こうが結果は同じことです。LOGOは，このプログラムを記憶する際に自動的に空白を調節するからです。

MZ-LOGOでは1度にひとつの手続きしか定義できません。すなわち，エディタの中で，toとendを書き加えて

写真1　2つの手続の定義



このように1度にふたつの手続きを定義しようとしても…



2番目の手続きは定義されていません。

```
to a :x          (初めからある行)
print :x
end              (初めからある行)
to b :x          (書き加えた行)
print :x :x
end              (書き加えた行)
```

とした場合，bという手続きは定義されません。

次に，エディット用のコマンドですが，これは，[↑]，[↓]，[←]，[→]の他はすべて，[CLR HOME]を押しながら[A]，[B]，…を押さなければなりません。1度に2つのキーを押すのは面倒だという人のために便利なやり方があります。それは，ファンクションキーを使うことです。MZ-LOGOでは，ファンクションキーは特殊な値を返すようになっています。その機能を表わすと以下のようになります。

- [F1]…[CLR HOME]＋a
- [F2]…[CLR HOME]＋b　または　[←]
- [F3]…[CLR HOME]＋c
- [F4]…[CLR HOME]＋d
- [F5]…[CLR HOME]＋e
- [F6]…[CLR HOME]＋f　または　[→]
- [F7]…[CLR HOME]＋g
- [F8]…[CLR HOME]＋h　または　[INST DEL]
- [F9]…[CLR HOME]＋i
- [F10]…[CLR HOME]

したがってファンクションキーを使えばエディット・コマンドのほとんどがひとつのキーで代用できるのです（コマンドの説明はマニュアルを見てください）。ただし，これらのファンクションキーは，F8とF10以外はコマンド・レベルでは使用できないので注意してください。プログラムの実行を止めるためにF7を押して何の効果もありません。

あと，プログラムに関して注意することは，なるべく" を使わない，ということです。これは，MZ-LOGOのほうに問題があると思うのですが，なぜか" が ["] に変わってしまうことがあります。たとえば，プログラム中に

```
make "x :x+1
```

という文があるとき

```
make [" x] :x+1
```

と，ときどき変わってしまうのです。こうなるともはやプログラムは動きません。エディタで正しく書き直してください。このようなことに神経を使うのは考えものですが，LOGOに動いてもらうためにはしかたありません。

4　" と：の考え方

LOGOでは名前の前に" とか：を付けて文字定数や変数を表わしています。

```
make "lis fput "x :lis
```

などとする場合，"lisと:lisは同じものです。このようなとき，" と：の使いわけはどうなっているのか疑問がわきます。このことについてちょっと考えてみましょう。その場合，名前ということが大事になってきます。名前はひとつの単語で表わされます。つまり

```
korewa.namae
```

は名前といえます。しかし，この名前はいったい何を表わすのでしょうか。手続きでしょうか，文字定数でしょうか，それとも変数でしょうか。実は，そのうちのどれにでもなり得るのです。これらのことをはっきりさせるために以下のように考えるとよいでしょう。しかしこれは筆者の独断で，その真偽はわかりませんが，便宜的な理解に役立つと思います。

LOGOは，toコマンドやmakeコマンドによって名前が与えられると，それらに記憶領域上のひとつの場所を割り当てます。そして，以後，その場所がどこかを知るためにその名前を使うのです。同じ名前が与えられたときは，いつも同じ場所を示すことになります。ある名前に対して与えられた場所には，その名前の内容が格納されています。この内容とは，手続き名に対してはプログラム，変数名に対してはその値のことです。文字定数に関しては何もないという情報が入っています。この時点では，その持っている内容はともかく，手続きや変数の名前に何の区別もありません。

LOGOは，たとえば

```
korewa.namae
```

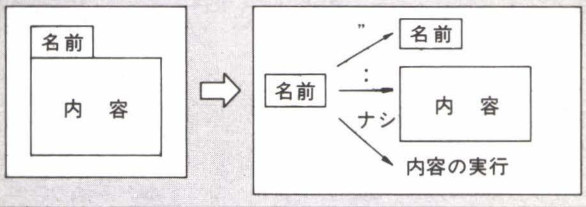
という名前に行き当ると，korewa.namaeが示している場所の内容を実行しようとします。もし，korewa.namaeが定義されている手続き名ならそのまま実行されます。しかし，それ以外のときはエラーとなってしまいます。理由は，その名前の示している場所の内容が，プログラムとして実行不可能だからなのです。そして，プログラムではないものを誤って実行しないようにLOGOに教える目印が，" や：なのです。

では" と：の違いは何でしょうか。結論をいえば，" は名前そのもの，：は名前の示す内容を値とする関数です（図1参照）。

たとえば

```
to square
```

図1 "と：の機能




```
    repeat 4 [fd 50 rt 90]
  end
```

と，squareという手続きが定義されているとき

```
  print square
```

は正方形を描いたあと，squareは値を返さないので空リスト〔 〕をプリントします。

```
  print "square
```

は，squareという文字列をプリントします。

```
  print :square
```

は，squareの内容

```
  [][repeat 4. [fd 50. rt 90.]]
```

をプリントします。この点，:squareはtext "squareと同じであるといえます。またこれはthing "squareとも等しくなっています。

```
  equalp (text "square)(thing "square)
  Result : true
```

しかし,文法上からはプログラムの内容を与えるのがtext,変数の内容を与えるのがthingとなっています。

以上で，"と：について理解できたでしょうか。少々理解できなくても，気分だけはわかってもらえたらと思います。繰り返しますが，"はそのまま，：は内容です。こんなちょっとしたことでも覚えておくと，少しでもLOGOが理解しやすくなるのではないでしょうか。

5 oblistの活用法

LOGOにはoblistというコマンド(?)があります。oblistはその名のとおりリストですから，コマンドというよりも変数名といったほうがよいかもしれません。oblistの内容を知るにには：を用いて

```
  print :oblist
```

などとします。このときプリントされる内容は，今までにLOGOが出会ったすべての名前(数は除きます)のリストです(図2参照)。このとき最後の136個はプリミティブ(基本命令)の名前になっています。そして，以後LOGOは新しい名前(すなわちoblistにない名前）に出会うごとに,それをoblistの最初に付け加えていくのです。(なお余談ですが，oblistの他にも,true,falseコマンドも実際に使用するときはその前に：を付けて使用します。true,falseはそれぞれ，常にtrue,falseという値を返します。LOGOのプリミティブのうち：を付けて使用するコマンドはこの3つだけなので，ここに書いておきます)。

さて,このoblistを使って何ができるでしょうか。まず考えられるのは，今までに定義したすべての手続き名をプリントすることです。こういうプログラムがあれば,今までにどんな手続きを定義したか知るのに便利です。そのためには,oblistから手続き名だけを抜き出してプリントすればいいでしょう。代わりに，definedpを使えればよいのですが…残念ながら現在市販されているMZ-LOGOではdefinedpが正常に機能しません。それで，意図するプログラムを書くことができないのですが，参考までに図3にプログラムを示しておきます。definedpを使わない場合は自分で内容を見ながら，それが手続きかどうか判断するしかありません。手続きの内容を知るには，printout(省略形po)があります。これは,

```
  printout square
```

図2 OBLISTの初期状態

```
> print :oblist
integer word namep keyp definedp listp memberp numberp wordp emptyp readline rc
readchar .deposit .examine out in .call char tempo music ascii toplevel random e
rall ern er erase save text thing poff pon : at setbg background px penreverse p
e penerase pc pencolor dot home cleartext cs clearscreen clean fence heading sho
wnp window wrap pos ycor xcor scrunch setpc setscrunch bk back ht hideturtle st
showturtle pd pendown pu penup seth setheading lt left rt right fd forward paint
 printout po type pr print close open false true round cos sin sqrt not or and e
qualp remainder quotient product sum lput fput sentence list count item last but
last butfirst first run = >.< / * - + " ) end repeat stop output to ed edit bye
if define load name make mem recycle oblist
>
```

図3 POTS と POTS1

```
to pots
  pots1 :oblist
end
```

```
to pots1 :lis
  if (count : lis) <  137. [stop]
  if definedp (first : lis) [print first : lis]
  pots1 butfirst :lis
end
```

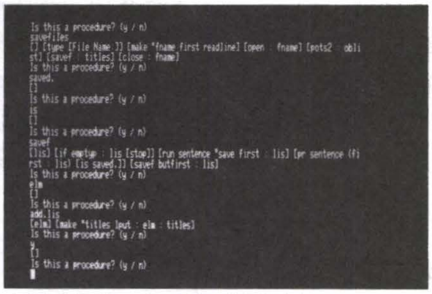
図4 **POTILES、POTS2、ADD. LIS**

```
to potitles
  pots2 :oblist
  print :titles
end

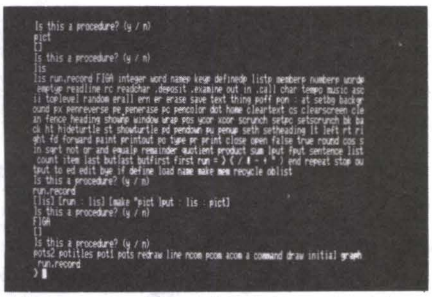
to pots2 :lis
  if (count : lis) <  137. [stop]
  print first :lis
  print text first :lis
  print [Is this a procedure? (y / n)]
  if (ascii readchar) =  121. [add.lis first : lis]
  pots2 butfirst :lis
end

to add.lis :elm
  make "titles lput :elm :titles
end
```

写真2　TOTITLESの使用例



内容をみながらyかnのキーを押します。



終わるとyのキーを押した名前がプリントされます。

という具合に使います。squareが前節で定義したものなら画面に

to square

repeat 4. 〔*fd 50. rt 90.*

end

とプリントされます。しかし，printoutの引数は手続き名でなければいけません。今の場合，内容を知りたい名前が手続きか変数かわかっていないので，出力は汚くなりますが，textコマンドを使うことにします。textコマンドは，前節でもちょっと触れましたが，手続き（や変数）の内容をリストにして返してくるものです。引数が手続きの場合，(実際は手続きに対してだけ使うのがノーマルな方法ですが）第1要素は引数のリストになっています。またそのとき引数名に：は付いていません。たとえば，

```
to add :x :y
  output :x +:y
end
```

という手続きがあるとき

```
print text "add
```

とすると，画面に

〔*x y*〕〔*output :x +:y*〕

とプリントされます。以上のことを知った上で図4のようなプログラムを作ってみました。これを実行すると，コンピュータがすべての名前とその内容をプリントしながら「これは手続きか？」と聞いてきますから，もし手続きならⓎのキー(小文字のモード)を押します。すると，次の名前とその内容をプリントしてきますから，上の操作を続けます。そして，すべての名前について調べが済むと，titlesという名前のリストにすべての手続き名が入っているので，それをプリントします（写真2参照)。

odlistにはまだ他の使い方があるかもしれません。しかし筆者はこれくらいしか考えつかないので，誰か面白い使い方を考えた人はぜひ教えてください。

6 プログラムのセーブとロード

前節ですべての手続き名をプリントするプログラム(potitles)を紹介しました。実は，これはセーブの手間を省くために考えたものなのです。MZ-LOGOのテープへのセーブの方法は実に面倒です。つまり，

open "ファイル名

で，ファイルをオープンした後

save 手続き名1

save 手続き名2

………………………

という具合に手続きをセーブしていき，最後に

close "ファイル名（open時と同じもの）

でファイルを閉じて完了です。ここで不便なのは手続き名をひとつひとつ覚えておかなければならないということです。このとき手続き名のリストがあれば，大事な手続きをセーブし忘れた，なんてことはなくなるでしょう。potitlesを実行した後には，

図5 **SAVEF**

```
to savef :lis
  if emptyp :lis [stop]
  run sentence "save first :lis
  pr sentence (first : lis) [is saved.]
  savef butfirst :lis
end
```

（注） run first :lis ではエラーとなるので runコマンドを使用しています。

titlesという名前のリストに手続き名が入っていますから，それを見ながらセーブしていけばよいわけです。しかし，セーブをもっと楽にするために図5のようなプログラムを考えます。これは，セーブすべき手続き名のリストを引数に与えてやれば，それらを自動的にセーブするものです。たとえば

open "EX1

でEX1という名前でファイルをオープンしたあと

savef [maru sankaku batsu]

とすると

maru is saved.

sankaku is saved.

batsu is saved.

とメッセージが出され，maru, sankaku, batsu，の3つ，の手続き（こういう名前の手続きがあると仮定して）がセーブされます。そしてそのあとに

close "EX1

としてやればよいのです。もっと実際的なsavefの使い方は，potitlesの実行中に，セーブしたくない手続きの場合はⓃのキー（実はⓎ以外ならなんでもよい）を押しておき，その実行が終わったあとに

savef :titles

とすることです。これらのことを考えると図6のようなプログラムもできます。これを使えば，最初にファイル名を与えて，あとはⓎのキーを押すだけでその手続きをセーブすることができます。

図6 SAVEFILES

```
to savefiles
  type [File Name:]
  make "fname first readline
  open :fname
  pots2 :oblist
  savef :titles
  close :fname
end
```

プログラムのロードは簡単です。最初に見つかったファイルをロードするには

load

だけでよいし，特定のファイルをロードするには

load "ファイル名

とすればよいのです。先の例でいえば

load "EX1

とすると，maru, sankaku, batsuの3つの手続きがロードされます。なお，同じ名前の手続きがワークスペース内にあるときは，前の手続きが消されますから注意しましょう。

7 LOGOによる音楽演奏

MZ-LOGOにはmusicという，音を鳴らす関数があります。これはBASICのMUSICとまったく同じ機能を持っています(BASICのマニュアルを見てください）が，実際に使用するときには工夫が必要です。

music "-A-BCDEFGA

と入力するとエラーになります。これはワードの中に使われている－が，ワードを分ける働きをするからです。つまり，まともにやったのでは，－や＋を含んだひとつのワードを作ることはできないのです。そこでwordを使うことにします。wordは2つのワードを引数とし，それらをつなげたひとつのワードを値として返します。

wordを使って

music word(word(word "- "A)"-) "BCDEFGA

とすると、上の例を正しく演奏します。しかし，－や＋が出てくるたびにwordを使っていたのではウンザリしてきますね。そこで演奏すべきデータ（musicへの引数）を作るプログラムを作ってみました。図7を見てください。このmdataというプログラムは，リストで与えられたデータをひとつのワードにして，前からあるmd1という変数にそれを付け加えていくものです。

make "md1 "　（md1の初期化）

Result:

mdata[-A-BCDEFGA]

（この時点でのmd1の値は"－A－BCDEFGA)

mdata[+B+C]

（この時点でのmd1の値は"－A－BCDEFGA＋B＋C)

上記のように，mdataを実行するごとにmd1の値であるデータの長さが長くなっていきますから，新たにデータを作る場合は，再びmakeコマンドでmd1に空ワード「"」を与える必要があります。mdataで作ったmd1というデータを実行するには

musik :md1

とすればよいのですから，少しは音楽を演奏させることが楽になったはずです。

図8のプログラムは，mdataで音楽データを作るとき，md1の初期化も同時に行なうものです。このmdata1の使い方は

mdata1 [-A-BC]

としたあと

make "m1 :md1

とすれば，m1という変数に"-A-BCというデータが移されます。それで次に

mdata [#C#D#E]

としたとき，前のデータをm1という名前で残しておくことができます。mdata1の後のmakeコマンドで，m1以外にもm2, m3, ……とデータを作って記憶させておけば，

図7 MDATA

```
to mdata :lis
  if emptyp :lis [stop]
  if (or (equalp first : lis "-) (equalp first : lis "+)) [make "md1 word : md1
(char (ascii first : lis))] [make "md1 word : md1 first : lis]
  mdata butfirst :lis
end
```

図8 **MDATA I**

```
to mdata1 :lis
  make "md1 "
  mdata :lis
end
```

図9 **M・GO**

```
to m.go :lis
  if emptyp :lis [stop]
  music thing (first : lis)
  m.go butfirst :lis
end
```

図10 **MUSIC. DATA**

```
to musik.data
  mdata1 [#C3#C#CD#C5#C3#C - B - B - B - B - A5 - B3#C3]
  make "m1 :md1
  make "m2 "#C7R
  mdata1 [#C3#F5#G#C3#C#C5R5#C3 - B - A - B - B - A - #G5 - #G8R5]
  make "m3 :md1
  make "m4 "#C8R5
  mdata1 [#C3 - B - A - #G - #G - #G]
  mdata [- A - B - A - #G - #G - #F]
  mdata [- #E5 - #G - #F9 - #F7R7]
  make "m5 :md1
  mdata1 [- B5 - A3 - #F - #F#C - B - A - B5 - A3 - #F - #F5R5]
  make "m6 :md1
  mdata1 [- A3 - B5 - A3 - B5#C - #F8R5]
  make "m7 :md1
  mdata1 [- B3#C - B - A - #F5R5]
  make "m8 :md1
  mdata1 [#C3D#C - B#C5D#C9#C5R5]
  make "m9 :md1
  make "m10 "#C5#C
  make "m11 "#F5#G3A
  make "m12 "A5
  make "m13 "#C3#C
  make "m14 "A7
  make "m15 "#G3A#G#F#E5#G
  make "m16 "#F7R3
  make "m17 "B3A#GA#G#F#E5
  make "m18 "#E7
  make "m19 "#F8
  make "m20 "R3
  mdata1 [#C3D - B#C4#C3#F7D5E3#F#F7#E5#F3#G#F9#F8R5]
  make "m21 :md1
end
```

日本音楽著作権協会承認第8352782号

図11 **MUSIC**

```
to music
  m.go [m1 m2 m3 m1 m4 m5 m6 m7 m8 m8 m9 m10 m11 m12 m13]
  m.go [m11 m14 m15 m16 m13 m11 m12 m13 m11 m14 m17 m18 m10]
  m.go [m11 m12 m13 m11 m14 m15 m19 m13 m11 m12 m13 m11 m14]
  m.go [m17 m20 m21]
end
```

```
music :m1
music :m2
music :m3
......................
```

とすることで，音楽を続けて演奏することができるので便利です。

次にいよいよ，**mdata1**と**make**コマンドで作られたデータを演奏させるプログラムを書きます。いま，

```
m1,m2,m3…m10
```

という名前の変数に**mdata1**で作られたデータが入っているものとします。これらを続けて演奏させるプログラムなら

```
to music.go
  music :m1
  music :m2
  ......................
  music :m10
end
```

とすればよいのですが，これでは面白くありません。そこで図9のようなプログラムはどうでしょうか。**m.go**はリストで与えられた変数名の値であるデータを順番に演奏していきます。今の例では，上と同じプログラムを

```
to music.go
  m.go[m1 m2 m3……m10]
end
```

と1行で書くことができます。

ところで，何もしないと演奏速度はModerato（中庸）となっています。これを変えるには**tempo**コマンドを使います。これもBASICと同じで，たとえば

```
tempo 1
```

とすると演奏速度がもっとも遅くなります。もっとも速くするときは

```
tempo 7
```

とします。

話ばかりで，具体例がなければつまらないので，実際に曲を演奏するプログラムを作ってみました。これは，中森明菜の「禁区」を演奏するプログラムです。tempoは５ぐらいが適当でしょう。これを図10，11に表わしました。図10がデータを作るmusic data，図11がそれを演奏するmusicです。実際に演奏させるためには

```
musik.data
```

と入力したあと

```
musik
```

と入力します。

8 色つきのグラフィック

９月号でも触れましたが，LOGOは色付きの絵を描くことができます。色に関するプリミティブはsetpc.setbg.pencolor（省略形pc），penerase（省略形pe），penreverse（省略形px）の５つです。

setpcはタートルの軌跡の色を指定するコマンドです。使い方は

```
setpc 3
```

とします。引数のところの数字は表１のような意味があります。この数字はsetbg，pencolorに対しても同じ意味を持ちます。また，引数に与える数値が０～７の範囲を出るときは，それらを８で割った余りとなります。すなわち８は０，９は１，(－３)は５，……という具合です。ただし(－１)に対しては，後に説明するpenreverseと同じ働きになるので注意してください。上記の例は，タートルの軌跡の色を紫色にしろ，という意味になります。また，この色は次にsetpcが実行されるまで保たれます。

setbgは画面の背景の色を指定するコマンドです。使い方はsetpcと同じく

```
setbg 6
```

とします。これは，背景の色を黄色にしろ，という意味になります。

pencolorは現在のタートルの軌跡の色に対応する数値を返す関数です。たとえば

```
setpc pc+1
```

とすれば，現在のタートルの軌跡の色の次の色が，以後の軌跡の色になります。

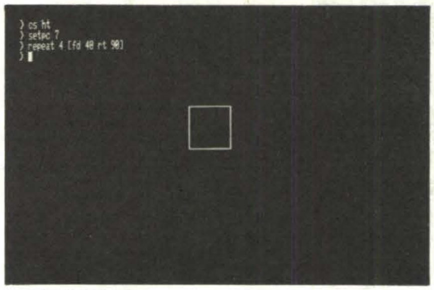
peneraseは，タートルが以前に描いた軌跡の上を通過すると，その部分が消えるようにするコマンドです。これは，実際的にはsetpc 0と同じです。(pcの値が0となっています）つまり０の色は黒というより消しゴムみたいなものと考えたほうがよいのです。

```
clearscreen hideturtle
setpc 7
repeat 4 [fd 40 rt 90]
penerase
repeat 4 [fd 80 rt 90]
```

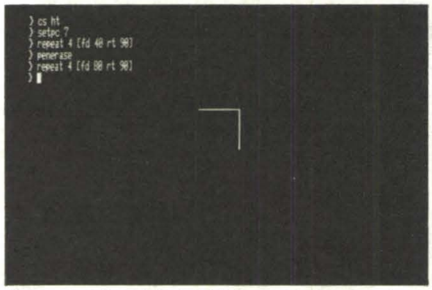
と，キーボードから入力してみてください。２度目にタートルがとおった部分が消されてしまいます（写真３参照)。なお，penereaseは次にsetpcが実行されてpcの値が0以外になるまで有効です。

penreverseは，タートルが以前に描いた軌跡を通過すると，

写真３　PENERASEの例

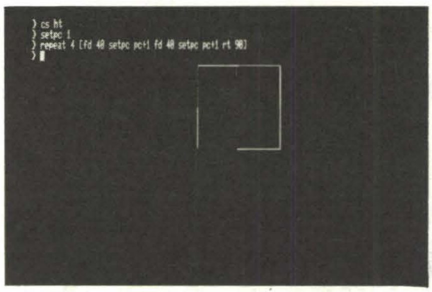


正方形を描いて…

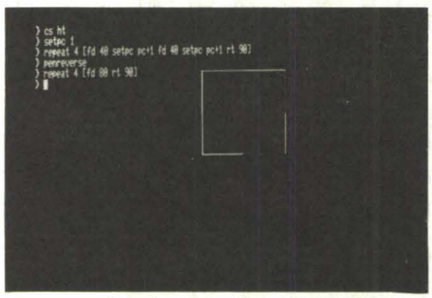


さらに大きな正方形を描くと，２度目にタートルが通った部分が消えます。

写真４　PENREVERSEの例



８種類の色で正方形を描いて…



penrevereの後，その正方形をなぞると色が反転します。

図12　RUN. RECORD

```
to run.record :lis
  run :lis
  make "pict lput :lis :pict
end
```

表１　数字と色の対応

0……黒	4……緑
1……青	5……水色
2……赤	6……黄
3……紫	7……白

表２　色の反転

黒（0）	←→	白（7）
青（1）	←→	黄（6）
赤（2）	←→	水色（5）
紫（3）	←→	緑（4）

その部分の色が反転するようにするコマンドです。色の反転は表2に示しています。また，何も描かれていない部分をとおるときの軌跡の色は白となります。penreverseはsetpc(−1)と同じ（実際，pcの値が(−1)となっています）ですから，(−1)を8で割ったあまりが7（白）であることを考えれば明らかでしょう。例として，次のように入力してみてください。

```
clearscreen hideturtle
setpc 1
repeat 4 [fd 40 setpc pc+1 fd 40 setpc pc+1
 rt 90]
penreverse
repeat 4 [fd 80 rt 90]
```

これによって，8色の色がどのように反転するかを目で確かめることができます（写真4参照）。なお，penreverseはpeneraseと同様，次にsetpcが実行されてpcの値が(−1)以外に変わるまで有効です。

上で

`clearscreen hideturtle`（省略形cs ht）

というコマンドを使用していますが，これは画面をクリアしてタートルのマークを画面から消すというコマンドです。タートルに新しい図を描かせる前は，これらのコマンドを実行することが多いようです。タートルのマークを消すには，それがないほうが図を描く速度がずっと速いからです。

以上の予備知識（色に無関係なグラフィックコマンドは既知とします）をもとにカラーの絵を描いてみましょう。といっても，座標を計算しながら描くのではBASICと同じで面白くありません。そこで，絵を描くときにタートルに与えた命令をすべて記憶しておくようなプログラムを考えます。このときに必要になるのがrunコマンドです。これは，引数として与えられたコマンドのリストを実行するものです。たとえば，

```
run [repeat 3 [fd 50 rt 120]]
```

とすればちゃんと，一辺が50の正三角形を描いてくれます。このrunコマンドを使って書いたプログラムが，図12のrun.recordです。これは，引数にコマンドのリストが与えられるごとにそのコマンドを実行し，pictという名のリストにそのコマンドを付け加えていく手続きです。たとえば

```
run.record [cs ht]
run.record [setpc 3]
run.record [repeat 4 [fd 50 rt 90]]
```

とすると画面上に紫色で正方形が描かれます。このとき変数pictの内容は

```
[[cs ht][setpc  3.][repeat 4.[fd 50. rt 90.]]]
```

というリストになっています。もしこの記録を手続きとして残しておきたいなら，defineコマンドを使うことで簡単に実現できます。これは，手続きをリストの形で定義するためのコマンドです。手続きは，もともとコマンドのリストからなるリストでした（67ページ，4参照）。手続きと普通のリストの違いは，その第1要素が引数のリストであるかどうかということだけです。つまり手続きであるリストの構造は

[[引数名1　引数名2…][コマンド][コマンド]…[コマンド]]

となっています。このような構造のリストに名前を付けて手続きにしてしまおう，というのがdefineコマンドなのです。たとえば

```
define "wa [[x y][output :x+:y]]
```

とすれば，ふたつの引数の和を求める手続きwaが定義できることになります。引数を含むリストの中の引数名の前には：はつきません。また，endもありません。これが，toによって手続きを定義する場合との大きな違いです。そこで手続きrun.recordから作られたpictを，手続きに変えるためには，pictの最初に，引数のリスト（今の場合引数がないので空リスト）を付け加えたものをdefineコマンドに与えればよいことになります。新しく作る手続きの名前をpict1とするときは

```
define "pict1 fput [] :pict
```

とすればよいのです。

図13のようなプログラムを考えます。これは，キーボードから1度にひとつだけのキーを押すことで絵を描き，それを記憶しておこうというものです。押すべきキーの意味は表3に示しておきました。図12のcommandという手続きの中では"を多用しています。これを見て，66ページ，3でいったこと（"を多く使用しないこと）と違っていると怒らないでください。これは，キーとコマンドの対応をわかりやすくするためなのです。たとえば

```
if (ascii :com)=102 [……]
```

と書くより

```
if equalp :com "f [……]
```

と書く方が，見たときわかりやすいからです。

絵を再現するためには，わざわざそれを手続きとして保存しておかなくても（テープにセーブする場合は手続きにしておかなければなりません）図14のプログラムで間に合わせることもできます。

```
redraw :pict
```

とするだけでよいのです（写真5参照）。図11〜13までのプログラムは，先月号に参考として載った，instant，command，……などの改良(?)版です。先月号で混乱をした人がいたらゴメンなさい。

図13　GRAPH

```
    to graph
      initial
      draw
    end

    to command
      make "com readchar
      if equalp :com "f [run.record [fd  2.]]
      if equalp :com "b [run.record [bk  2.]]
      if equalp :com "u [run.record [pu]]
      if equalp :com "d [run.record [pd]]
```

```
    if equalp :com "r [run.record [rt  30.]]
    if equalp :com "l [run.record [lt  30.]]
    if equalp :com "h [run.record [home]]
    if equalp :com "c [run.record [clean]]
    if equalp :com "a [acom]
    if equalp :com "p [pcom]
    if equalp :com "g [gcom]
    if equalp :com "x [run.record [px]]
    if equalp :com "e [run.record [pe]]
    if equalp :com "n [ncom]
    if equalp :com "q [toplevel]
    if equalp :com "i [run.record [repeat  90. [fd  1. rt  1.]]]
    if equalp :com "j [run.record [repeat  90. [fd  1. lt  1.]]]
    if equalp :com "k [line]
    if equalp :com "v [st]
    if equalp :com "w [ht]
  end

  to acom
    pr [position?]
    run.record sentence "at readline
  end

  to gcom
    pr [back ground color?]
    run.record sentence "setbg ((ascii readchar) -  48.)
  end

  to pcom
    pr [pen color?]
    run.record sentence "setpc ((ascii readchar) -  48.)
  end

  to line
    pr [line length?]
    run.record sentence "fd (first readline)
  end

  to ncom
    make "pict butlast :pict
    clearscreen
    hideturtle
    redraw :pict
    draw
  end

  to initial
    window
    setscrunch  8.00000N01
    cs
    cleartext
  end

  to draw
    command
    draw
  end
```

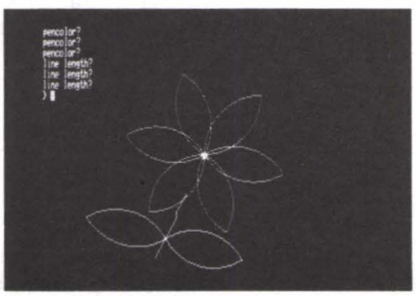
図14 **REDRAW**

```
to redraw :lis
  if emptyp :lis [stop]
  run first :lis
  redraw butfirst :lis
end
```

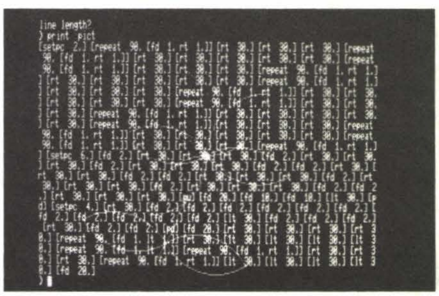
表3 COMMANDに対するキーの意味

キー	意　　味	キー	意　　味
f	2歩前進。	x	penreverse
b	2歩後退。	e	penerase
u	penup	n	ひとつ前のキー入力をとり消す。
d	pendown	q	絵を描くのをやめる。
r	30度の右回転。	i	右廻りに4分の1円を描く。
l	30度の左回転。	j	左廻りに4分の1円を描く。
c	タートルの状態を変えず画面クリア。	k	指定する歩数だけ前進。
a	タートルを指定する座標へ移す。	h	タートルを座標(0, 0)へもどす。
p	ペンの色を変える。	v	タートルのマークを表わす。
g	背景の色を変える。	w	タートルのマークを消す。

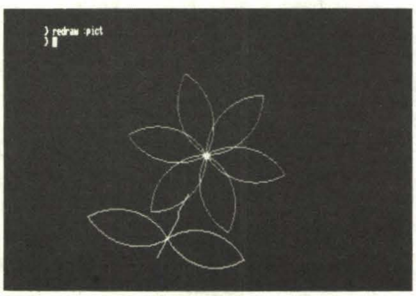
写真5　絵を描いてそれを再現する



graphによって絵を描くと



pictの内容はこのようになり…



redrawでそれを再現できます。

9 プリンタへの出力

MZ-LOGOはプリンタへ出力することが可能です。そのためにはまず

```
pon
```

というコマンドを実行させます。このコマンドには引数はありません。実行後は画面と同じ出力が(エラーメッセージに到るまで)プリンタへも出力されます。

プリンタへの出力を止める場合は,

```
poff
```

というコマンドを実行させます。このとき以後の出力は画面だけにされることになります。またこのコマンドにも引数はありません。

例として,72ページ,8で示したcommandというプログラムのリストをプリンタへ出力することを考えましょう。この場合は

```
pon
printout command
poff
```

とすればよいのです。

pon,poffに関して,ただひとつ注意すべきことは,ponを実行させてからpoffを実行させる前に,エディット・モードに移ることです。これは,エディタのコントロールキャラクタまでがプリンタへ出力されて,とても読みにくいものとなるからです。それに,ただでさえ反応の鈍いエディタの反応速度がさらに遅くなってしまいます。

10 8月号を振り返って

本誌8月号の特集の時点では,MZ-LOGOが完成していなかったため(V.1.0にはまだバグが少しいるようですが),その記事の中で使われているLOGOはMZ-LOGOと多少異なっています。大部分はApple用のTerrapin LOGOを参考にして書かれているようですので,ここでTerrapin LOGOとMZ-LOGOの違いについて説明してみたいと思います。なお,余談ですが,市販されているMZ-LOGOはApple用のあまり知られていない(すなわち,日本で手に入りにくい)ほうのApple LOGOを見本にしているようです(別にApple LOGOの機能が,Terrapin LOGOに劣るといってるわけではありませんので,悪しからず)。

Terrapin LOGOやApple LOGOとMZ-LOGOの大きな違いは,前者2つのLOGOのプリミティブは英字の大文字であるのに対し,MZは小文字であるという点です。したがって,8月号の記事の中のプログラムも,そのプリミティブを小文字に直すだけで,そのままMZ-LOGOで動くものが少なくありません。その際注意すべきことは,ifコマンドの形式がTerrapinとMZとでは異なるということです。つまり,Terrapin LOGOで

```
IF 条件 THEN 文1
```

あるいは

```
IF 条件 文1
```

と書くべきものは,MZ-LOGOでは

```
if 条件 [文1]
```

となります。またTerrapin LOGOの

```
IF 条件 THEN 文1 ELSE 文2
```

に対しては,MZ-LOGOでは

```
if 条件 [文1] [文2]
```

となります。以上の2つの注意事項を知っていれば,TerrapjnからMZへ,あるいはMZからTerrapinへプログラムを移植することは簡単なことでしょう。ただし,Terrapjn LOGOとMZ-LOGOでは機能が同じでも名前の異なるものや,名前が同じでも機能の異なるプリミティブがあり混乱を招きやすくなっています。そこで参考までに,Terrapin LOGOとMZ-LOGOのプリミティブの対応表を作成してみましたので掲載しておきます。

表4 TerrapinとMZの対応表

グラフィック コマンド

MZ-LOGO	TerrapinLOGO	意味
back(bk)	BACK(BK)	タートルを後退させる。
setbg	BACKGROUND(BG)	画面の背景の色を指定する。
clean	CLEARSCREEN(CS)	画面をクリアする。
clearscreen(cs)	DRAW	画面がクリアされ,初期化される。
forward(fd)	FORWARD(FD)	タートルを前進させる。
heading	HEADING	タートルの向いている方向を与える。
hideturtle(ht)	HIDETURTLE(HT)	タートルのマークを画面から消す。
home	HOME	タートルを初期状態に設定する。
left(lt)	LEFT(LT)	タートルを反時計方向に回転させる。
fence	NOWRAP	タートルが画面から出たときエラーとする。
setpc	PENCOLOR(PC)	タートルの軌跡の色を指定する。
pendown(pd)	PENDOWN(PD)	タートルの軌跡を残すようにする。
penup(pu)	PENUP(PU)	タートルの軌跡が残らないようにする。
right(rt)	RIGHT(RT)	タートルを時計方向に回転させる。
setheading(seth)	SETHEADING(SETH)	タートルの方向を設定する。
at	SETXY	指定した座標へタートルを移動させる。
showturtle(st)	SHOWTURTLE(ST)	タートルのマークを画面に出す。
wrap	WRAP	タートルが画面から出たとき反対側に出す。
xcor	XCOR	タートルの現在の水平座標を与える。
ycor	YCOR	タートルの現在の垂直座標を与える。
background	LAST BL TS	画面の背景の色を与える。
pencolor(pc)	LAST TS	タートルの軌跡の色を与える。
shownp	FIRST BF TS	タートルのマークが画面に出ているかどうか。
window		タートルが画面から出たとき表示しない。
dot		指定した座標へ点を打つ。
penerase(pe)		タートルが通過した部分の軌跡を消す。
penreverse(px)	PENCOLOR 6	タートルが通過した部分の軌跡の色を反転。
pos	LIST XCOR YCOR	タートルの現在の座標を与える。
	FULLSCREEN	フル・グラフィックスクリーンを与える。
	NODRAW(ND)	グラフィック・モードからテキスト・モードに移る。
	SETX	指定したx座標までタートルを水平に移動。
	SETY	指定したy座標までタートルを垂直に移動。
	SPILTSCREEN	グラフィック・モードでテキスト・スクリーンも見せる。
	TOWARDS	タートルの向くべき角度を与える。
	TURTLESTATE(TS)	タートルに関する情報を与える。

数値演算

MZ-LOGO	Terrapin LOGO	意味
+	+	たし算。
-	-	ひき算。
*	*	かけ算。
/	/	わり算。
cos	COS	余弦。
integer	INTEGER	小数部分の切り捨て。
quotient	QUOTIENT	商。
random	RANDOM	整数の乱数。
remainder	REMAINDER	余り。
round	ROUND	もっとも近い整数。
sin	SIN	正弦。
sqrt	SQRT	平方根。
product		積。
sum		和。
	ATAN	逆正接。
	RANDOMIZE	乱数の初期化。

ワードとリストに関する演算

MZ-LOGO	Terrapin LOGO	意味
butfirst	BUTFIRST(BF)	最初以外の要素。
butlast	BUTLAST(BL)	最後以外の要素。
first	FIRST	最初の要素。
fput	FPUT	リストの最初に要素を付け加える。
last	LAST	最後の要素。
lput	LPUT	リストの最後に要素を付け加える。
list	LIST	リストを作る。
sentence	SENTENCE(SE)	引数をつなげてリストを作る。
word	WORD	ワードをつなげる。
count		リストの要素の個数を与える。
item		リストから要素を取り出す。

手続きの定義とエディット

MZ-LOGO	Terrapin LOGO	意味
define	DEFINE	手続きをリストから定義する。
edit(ed)	EDIT(ED)	手続きをエディットする。
end	END	手続きの終わりを示す。
to	TO	手続きの定義を開始する。

条件式

MZ-LOGO	Terrapin LOGO	意味
and	ALLOF	論理積。
or	ANYOF	論理和。
if	IF	条件判断。
not	NOT	否定。
	ELSE	IF…THEN…ELSEで使用。
	IFFALSE(IFF)	TESTの結果がFALSEのとき実行。
	IFTRUE(IFT)	TESTの結果がTRUEのとき実行。
	TEST	条件の判断を行なう。
	THEN	IF…THEN…ELSEで使用。

条件式に使われる述語

MZ-LOGO	Terrapin LOGO	意味
>	>	より小か？(数)
<	<	より大か？(数)
=	=	等しいか？(数)
equalp	=	等しいか？
listp	LIST?	リストか？
numberp	NUMBER?	数か？
namep	THING?	変数として定義されているか。
wordp	WORD?	ワードか？
emptyp	=",=[]	空ワードあるいは空リストか？
memberp		リストの要素にあるか？
definedp		手続きとして定義されているか？
true	"TRUE	常にtrueの値を与える。
false	"FALSE	常にfalseの値を与える。

手続きの実行の制御

MZ-LOGO	Terrapin LOGO	意味
output	OUTPUT(OP)	関数の値を返す。
repeat	REPEAT	繰り返し。
run	RUN	リスト中のコマンドを実行する。
stop	STOP	実行中の手続きの実行を止める。
toplevel	TOPLEVEL	すべての手続きの実行を停止し，入力待ちとなる。
	GO	ラベルへジャンプする。

入出力

MZ-LOGO	Terrapin LOGO	意味
ascii	ASCII	文字のアスキーコードを与える。
char	CHAR	指定したアスキーコードの文字を与える。
cleartext	CLEARTEXT	画面の文字をクリアする。
print(pr)	PRINT(PR)	画面への表示。
type	PRINT1	改行をせず画面への表示。
keyp	RC?	キーボードから入力があるか？
readchar(rc)	READCHARACTER(RC)	キャラクタ・バッファ内の最初に入力された文字
readline	REQUEST(RQ)	キーボードからの入力をリストで与える。
in		I/Oポート上の1バイトのデータを値として返す。
out		I/Oポート上に1バイトのデータを出力する。
pon	OUTDEV 1	以後の出力をプリンタへ出す。
poff	OUTDEV 0	プリンタへの出力を止める。
	CLEARINPUT	キャラクタ・バッファをクリアする。
	CURSOR	指定した位置へカーソルを移動させる。
	OUTDEV	データを出力すべきスロットを指定する。
	PADDLE	パドルのダイアルの値を返す。
	PADDLEBUTTON	パドルのボタンが押されているか調べる。

名前付け

MZ-LOGO	Terrapin LOGO	意味
make	MAKE	名前を付ける。
thing	THING	変数の内容を与える。
name		名前を付ける。

ワークスペースの管理

MZ-LOGO	Terrapin LOGO	意味
erase(er)	ERASE(ER)	指定した手続きを削除する。
erall	ERASE ALL(ER ALL)	すべての手続き，変数名を削除する。
ern	ERNAME	変数名を削除する。
bye	GOODBYE	LOGOから脱出する
printout(po)	PRINTOUT(PO)	手続きの内容を表示する。
load	READ	ファイル中の手続きをロードする。

save	SAVE	手続きをセーブする。
close		ファイルを閉じる。
open		ファイルを開く。
	CATALOG	ディスクにあるファイル名を表示する。
	ERASEFILE	ディスクからファイルを削除する。
	ERASEPICT	ディスクにあるピクチャーファイルを削除する。
	READPICT	ディスクからピクチャーファイルをロードする。
	SAVEPICT	ディスクに画面をセーブする。

デバッグ

MZ-LOGO	Terrapin LOGO	意　味
	CONTINUE(CO)	PAUSEで止められた実行を再開する
	PAUSE	実行を止める。
	NOTRACE	トレースをしなくする。
	TRACE	手続きをトレースする。

その他

MZ-LOGO	Terrapin LOGO	意　味
setscrunch	•ASPECT	垂直方向のスケールを設定する。
•call	•CALL	機械語の命令を実行する。
•deposit	•DEPOSIT	1バイトのデータの書き込み。
•examine	•EXAMTINE	1バイトのデータの書き出し。
recycle	•GCOLL	ゴミ集めを実行する。
mem	•NODES	現在の有効node，残りのnode数を表示する。
oblist		すべてのコマンド，手続きのリストを与える。
scrunch		垂直方向のスケール値を与える。
music		音を鳴らす。
tempo		音の長さを設定する。
	•BPT	システムのデバッグのためにモニタへ入る。
	DOS	DOSモニタへコマンドを送って実行する。

さて，8月号の特集記事のうちで，青木氏の記事(8月号：p.38～p.41）には興味深いプログラムがあります。中でも，特に面白そうな，「多倍長計算」と「電話帳」のプログラムをMZ-LOGOで動くように直してみましたので，これを図15，16に示しておきます。図15，16は先に述べた注意に従って，もとのプログラムを機械的にMZ-LOGOに移したものですが，図15の「多倍長計算」の中のplus1という手続きは記憶容量を使いすぎて動いてくれません。それを手直ししたものが図17のplus2です。plus1からplus2への変更に伴って，図15のplusという手続きも図17のplus.newのように変更します。その他の手続きは図15のままです。「多倍長計算」「電話帳」の実行結果を，それぞれ図18，19に示しておきます。これは，前節で述べたponのあとに手続きを実行させたときのプリンタへの出力です。

図15　多倍長計算

```
to plus1 :x :y :c
  if emptyp :y [output if (equalp : c [ 0.]) [: x] [plus1 : x [ 1.] [ 0.]]]
  if emptyp :x [output if (equalp : c [ 0.]) [: y] [plus1 : y [ 1.] [ 0.]]]
  if ((last : x) + (last : y) + (first : c)) <  10. [output lput ((last : x) + (
last : y) + (first : c)) plus1 butlast : x butlast : y [ 0.]]
  output lput ((last : x) + (last : y) + (first : c) -  10.) plus1 butlast :x bu
tlast :y [ 1.]
end

to plist :lis
  if emptyp :lis [stop]
  type remainder (first : lis)  10.
  plist butfirst :lis
end

to input :msg
  type :msg
  output readline
end

to plus
  make "m input [FIRST BIG NUMBER]
  make "n input [SECOND BIG NUMBER]
  plist plus1 :m :n [ 0.]
  print "
end
```

図16　電話帳

```
to insert :data :lis
  if emptyp :lis [output fput : data : lis]
  if compare (namae : data) (namae first : lis) [output fput : data : lis] [outp
ut fput first : lis insert : data butfirst : lis]
end

to compare :nam1 :nam2
  if (ascii first : nam1) < (ascii first : nam2) [output : true]
  if (ascii first : nam1) > (ascii first : nam2) [output : false]
  output compare butfirst :nam1 butfirst :nam2
end
```

```
to namae :data                          to denwabango :data
  output first :data                      output last :data
end                                     end

to search :nam :lis
  if emptyp :lis [output []]
  if equalp (namae first : lis) :nam [output first : lis]
  output search :nam butfirst :lis
end

to makedata
  make "element input [NAME & DATA]
  if emptyp :element [stop]
  make "denwacho insert :element :denwacho
  makedata
end

to tellme :nam
  make "ans search :nam :denwacho
  if emptyp :ans [print [NO DATA]] [print (sentence namae : ans "wa denwabango :
 ans)]
end
```

図17　多倍長計算の改良

```
to plus2 :x :y :c
  if (count : x) < (count : y) [make "plk : x make "x : y make "y : plk]
  if emptyp :y [output if : c =  0. [: x] [plus2 : x [ 1.]  0.]]
  make "plk (last : x) + (last : y) + :c
  if :plk <  10. [make "c  0.] [make "plk : plk -  10. make "c  1.]
  output lput :plk plus2 butlast :x butlast :y :c
end

to plus.new
  make "m input [FIRST BIG NUMBER]
  make "n input [SECOND BIG NUMBER]
  plist plus2 :m :n  0.
  print "
end
```

図18　多倍長計算のプリンタ出力

```
> plus.new
FIRST BIG NUMBER> 1 2 3 4 5
SECOND BIG NUMBER> 4 5 6 7 8
58023
> plus.new
FIRST BIG NUMBER>  2 3 4 5 6 7 8 9 0
SECOND BIG NUMBER> 3 4 5 6 7 8
234913568
```

図19　電話帳のプリンタ出力

```
> make "denwacho []
> makedata
NAME & DATA> Maekawa 1822
NAME & DATA> Aoki 2430
NAME & DATA> Ito 65 97
NAME & DATA> Matsuzaki 1234
NAME & DATA>
> tellme "Aoki
Aoki wa  2430.
Result:  2430.
```

11 おわりに

落ち穂ひろいという割に内容が多くなってしまいましたが，今回の記事，少しは参考になったでしょうか。確かに，MZ-LOGOには扱いにくい点があり，それが本格的にLOGOのプログラムを作る上での障害になっていることは否定できません。しかし，そんなところでつまづいていたのでは，LOGOの素晴らしさを知る前に「もう，やめた」なんてことになりかねません。以後もマニュアルに関して不十分な点を説明していきたいと思っておりますから，根気よくMZ-LOGOに付き合ってください。今月号，先月号，先々月号を合わせても，本文で解説されたコマンドは全コマンドの30%ぐらいしかありません。LOGOを完全に解説するのもまだまだ時間がかかりそうです。なお，参考までに今まで本文で解説されたコマンドを付表に示しておきます。

付表　本文中で説明されたコマンド（重複するものは除く）

入門タートルグラフィック(9月号)
forward　right　left　repeat　setpc　paint
入門MZ　LOGOとリスト処理(10月号)
first　butfirst　last　butlast　fput　lput　length emptyp　memberp　list　sentence
MZ LOGOあれこれ (今月号)
at　text　thing　definedp　printout　oblist save　load　open　close　music　tempo　word setbg　pc　penerase　penreverse　clearscreen hideturtle　define　pon　poff

パソコンテレビX1がソフトだけで MZ-2000 MZ-2200に早変り!!

驚異

システムソフトウェアコンバータ

BASIC HOUSE

機種：X1(CZ-800C)　製品番号：B6-2213　価格：¥3,800

大容量43500バイトのユーザーエリア、豊富なソフトも使えます。

■史上初！システムソフトウェアコンバータ■

メモリが足りない。ソフトが足りない。もう、こんな言は言わせません。名機MZ-2000・MZ-2200用のBASIC、大フリーエリア、豊富なソフトのあるMZ-1Z001・MZ-1Z002(共にBASICテープの名称)がX1で使えるからです。これらのテープはそのままではX1で読み出す事も起動させることもできませんが、この不可能を可能にする—それがシステムソフトウェアコンバータです。

■操作はカンタン！購入直後に10分間だけお付き合い下さい■

システムソフトウェアコンバータは、MZ-1Z001(またはMZ-1Z002)をX1にて読み出し、X1のハードに合わせて内部を改造するプログラムです。改造が終われば(この部分は9秒で完了)テープにSAVEされます。もちろんこれらの手順は親切な対話式のため、マニュアルなしでも簡単に使えます。こうして一度、X1用MZ-1Z001、MZ-1Z002(名付けてCZ-1Z001、CZ-1Z002)を作ってしまえば、後は「CZ-8CB01」(付属BASIC)と全く同等に扱えます。CZ-8CB01を使えばX1に、CZ-1Z001を使えばMZ-2000・MZ-2200まさにソフトだけで2台分活用できます。

■MZ-2000、2200になったX1■

次からはCZ-8CB01と同じ要領で起動でき、ローディング時間も1分42秒に短縮されています。CZ-1Z001(CZ-1Z002)を使えば、MZ-2000・2200で作成されたBASICアプリケーションプログラムのLOAD、SAVE(キーボードからの入力ももちろん)、RUNが出来ます。当然文法の違いもないため手直しも不要です。またMZ-80B用ソフトも、MZ-2000で動くものであれば同様に使えます。(B用のゴルフゲーム、君はターゲット等もお使いになれます) キャラクタや、フリーエリア(43500バイト、MZ-1Z002は39148バイト)まで完全コンパチ、しかもA面、B面に同内容を入れて販売いたしますので、A面でCZ-1Z001を、B面でCZ-1Z002をといった高度な活用もできます。

※※※御注意※※※

- 本ソフトはMZ-1Z001(MZ-1Z002)をX1で動くようにするものです。著作権上、MZ-1Z001(¥2,150)かMZ-1Z002(¥7,000)を合わせて御購入下さい。なおどちらにするか迷った方は、MZ-1Z001にして下さい。
- マシン語レベルの操作があり、かつその中でI/O操作をしている場合はMZ-80B→MZ-2000の時と同様正常に作動しないことがあります。逆に、オールマシン語でモニタからLOADするものでも種類によっては動作するものがあります。
- ディスク版、倍精度＋進演算BASIC版、マシンランゲージ版、LOGOなども続々発売予定です。御期待下さい!!

各種BASICテキストコンバータ　絶賛発売中！

X1	PC-8001	→CZ-800	B6-1483	¥3,800
	PC-8801	→CZ-800	B6-1493	¥3,800
	MZ-80B・2000	→CZ-800B	B6-1413	¥3,800
	MZ-80K/C・1200	→CZ-800	B6-1433	¥4,200
MZ-700	PC-8001	→MZ-700	B5-1483	¥3,800
	PC-8801	→MZ-700	B5-1493	¥3,800
	X1用逆アセンブラ		B6-1400	¥4,200

開発室速報！　一新 発 売一

■パソコンテレビX1システムソフトウェアコンバーター　第2弾!! LOGO & PASCAL　製品番号：B6-2217　定価4,200

パソコンテレビX1で人工知能言語、タートルグラフィックスで思いどおりの図形が描ける、MZ-LOGOが動きます。またB面には話題の言語、PASCALのコンバートソフトが入っています。本ソフトは著作権上MZ-LOGO(¥9,800)とMZ-1Z004(¥12,000)が必要です。

■システムソフトウェアコンバーター　第3弾、第4弾、第5弾近日発売予定!!

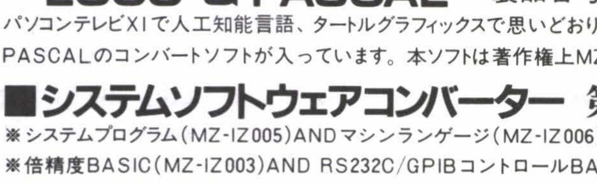
※システムプログラム(MZ-1Z005)ANDマシンランゲージ(MZ-1Z006)
※倍精度BASIC(MZ-1Z003)AND RS232C/GPIBコントロールBASIC(MZ-1Z010)
※DISK BASIC(MZ-2Z001)AND倍精度DISK BASIC(MZ-2Z003)

■特別企画 パソコンテレビX1が名機PC8001に…早変り！近日発売予定

注：著作権上当ソフトはX1とPC8001を持っている方に限ります。

お詫び：PC-6001→X1、X1↔MZ-700用のBASICテキストコンバータの開発が遅れており一部の方々に大変御迷惑をおかけしておりまことに申し分けございません。9月末には出荷できる予定でおります。

当社の製品はお近くのマイコンショップ等でお求め下さい。近くにない場合は、現金書留にて直接お申し込み下さい。



宇都宮の本格的なマイコンショップ BASIC HOUSE

〒320 宇都宮市桜3丁目2—17太平ビル
☎0286-33-1994　3F ㈱計 測 技 研
☎0286-37-3997　4F 栃木マイコン学院

X1 マシン語入門

システムソフトの活用

松木 透

これまで，X1というと,「メモリが少ない」,「ゲーム専用」,「マシン語も使えないくせに！」なーんてけなされていたことも結構あったようですが，よくよく考えてみれば，メモリ容量にせよ，ハードの性能にせよ，MZ-2200程度の能力はそのまま持ち合わせているうえに，PCGとかPSGといった高性能な付加機能を持っているんですね（ウソだと思ったらカタログをよく見てみよう！)。

それでは，なぜこんなことをいわれているのかというと，まず

CZ-8CB01(付属BASIC)のフリーエリアがたったの23Kである。

というのがあります。だいたい，こんなに大きなプログラムを作れる人はほとんどいないのに，と思うのですが，データ（たとえば住所録）ならそうとも限りません。プログラムが10Kバイトとすれば，データは残りの13Kです。MZでは43K－10K=33Kですから，2.5倍以上になっています。これでは，なるほど小さい。

ゲーム専用，というのは，X1にはPCG，PSG，ジョイスティック，拡大文字機能，パレット，プライオリティ…などなど，ゲームに使うと面白さが増す機能（もちろんゲーム以外の応用もできる）がたくさんあるため，そう見えてしまうのかも知れません。それに，ゲームソフトはかなりすごいのが（ハードがあれだけすごいのですから当然でしょうけど）多く，逆にビジネスソフトが「メモリが足りない」ために少ししかありません。専用，はいい過ぎですが，ある程度は当たっていると認めざるを得ません。

また，マシン語に関してはBASICに簡単な「マシン語モニタ」を備えているだけで，プログラムを作るのは極めて困難です。なぜなら，マシン語プログラムの作成には「アセンブラ」などの開発ツール（ソフト）が絶対に必要だからです。残念ながら，X1用のマシン語開発ツールは現在ひとつも発売されていません。ソフトメーカーの一部は社内専用に作って持っているところもあるらしいのですが，我々ユーザーに使うことはできません。マシン語モニタでは，他人の作ったマシン語プログラムを入力する程度しかできないのです。

幸いX1にはPCGがあるため，BASICだけでもある程度スピード感のあるゲームが作れます。が，ゲームの中でも「おっもしろい!!」と感じるものは，必ずといってよいほどマシン語が使われています。X1にとっても,「マシン語は関係ない！」では済まないのです。

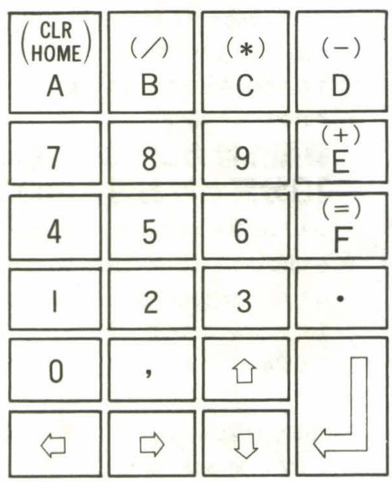
救いの神！登場!?

いままでいろいろと考えながらX1を使う日が続いたのですが，9月号のSOFT TOUCH（BASIC HOUSEのシステムソフトウェアコンバータ）を見て唖然としました。なんと，テープ1本でX1がMZ-2000/2200になってしまうというのです。私はいても立ってもおられず，早速マイコンショップに急ぎました。それと同時に，友人にMZ用のソフトをいくつか持ってきてくれるよう，頼みました。

実際にやってみて，再び驚きました。BASICだけのはもちろんのこと，POKEの入っているものや，USRの使われているものの多くも動いてしまったからです。OUT@文もみんな大丈夫でした。フリーエリアも43Kと，以前に買ってみたdB-BASICよりずっと広く，これならビジネスにも使えそうです。

数日後,「でも，これは無理だろう」と，友人が一本のテープを持ってきました。これが記事を書くきっかけになったのです。

図1　16進テンキーモード



彼が持ってきたのは，EA-MZというマシン語開発ツール(エディタアセンブラ)でした。もちろんオールマシン語です。MZと同様に，モニタからロードすると…何と動いてしまったのです。正にX1がMZに変身したようなものです。

これで3つ目の問題まですべて解決しました。かなり変則的な方法ですが，X1をMZとして使えばマシン語ソフトが開発できるというわけです。救いの神だけでなく，仏様やキリストまで来てくれてしまった!?という感じですね。

MZでマシン語を使う方法

MZでは，このほかに，IPLから直接ロードするマシン語関係のソフトがあります。一応全部見てみましょう。

①MZ-1Z006（マシンランゲージともいいます）

②MZ-1Z005(システムプログラムともいいます)

③EA-MZ

テープだけ（ディスクなし）の場合でも，3とおりあります。価格は①は7,000円，②は25,000円です。③はアスキー製で，9,800円です。

さきほど触れたように，X1でシステムソフトウェアコンバータ（B6-2213)を利用している人は，CZ-1Z001とCZ-1Z002のどちらでも，MZ-2000同様モニタから③をロードすれば使えます。

①，②については，IPLから動くため，CZ-1Z001などの力をおよぼすことができません。が，ごく最近になってシステムソフトウェアコンバータがシリーズ化されて，MZ-1Z005用などが発売されたために，これらもすべてX1で利用できるようになりました。

MZ-1Z006を使うには

MZ-1Z006は，簡単にいえばマシン語モニタを強化したような内容です。強化した部分は，レジスタ（BASICの変数のようなもの）の表示・エディット，ディスアセンブル，そのテープへの出力（ソースリスト化），ブロック転送，などです。

MZ-1Z006をX1で動かすには，X1，MZ-1Z006の他に，

システムソフトウェアコンバータ（MZ-1Z005・MZ-1Z006用）…定価 4,200円（型番B6-2218)

が必要です。これが1本あれば，あとで出てくるMZ-1Z005も動かせます。

コンバートの作業はとても簡単ですが，一度頭をクリーンにして読んでくださいね。

◇ 1 ◇

X1に付属のBASICから，MON⏎でモニタに入り，L⏎でB6-2218のB面（A面はMZ-1Z005用）をロードします。

◇ 2 ◇

ロードが終わると，テープが初めのところまで巻き戻り，フタが開きます（EJECTされます)。と同時に画面がクリアされ画面2のように表示されます。

◇ 3 ◇

MZ-1Z006のテープをセットして，フタを閉じると，自動的にロードしはじめます。5つのブロックに分かれているため，BASIC(MZ-1Z001など)のときより，やや時間がかかりますが，自動的に約7分ほどで終了し，画面3のように表示されます。

◇ 4 ◇

ここでもう一度，B6-2218（B面）をセットします。スペースキーを押すと，きちんとセットされたかどうか確認したうえで自動的にセーブが始まります。同時に，キーボードに手を触れないよう，メッセージが出ます。実際は触れても問題はないようですが，電源スイッチだけはやめておいたほうがよいです。

◇ 5 ◇

セーブが終わるとテープが巻き戻り，画面4のようになります。この時点で既にコンバート作業は終了しているので，カセットを取り出してしまってもよいのですが，一応IPLから起動してみましょう。電源スイッチを入れ直すか，何かキーを押す，（BOOTコマンドと同じになります)と，いつものIPL　is～ が表示され，やがてCZ-1Z006(MZ-1Z006のX1版，という意味。B6-2213を持っている人ならもうおわかりですよね）が起動します。

システムソフトウェアコンバータシリーズは，このようにしてコンバートする（重ね書きする）ため，ツメが折られていない状態で販売されています。一度これらの作業を済ませれば，あとはいつでも「CZ-8CB01」と同様にIPLから起動できますから，ツメは折っておきましょう。

IPLからCZ-1Z006を起動したあとの使い方ですが，これはMZでIPLからMZ-1Z006を起動したあととまったく同じです。本誌8月号の89ページには紹介記事がありましたので，持っている人は参考にするとよいでしょう。

もちろんMZ-1Z006には120ページに渡るマニュアルが付属になっていますから，

画面1　B6-2218(B面)をロード中

```
      SHARP-HuBASIC CZ-8CB01 V1.0
   Copyright (C) 1982 by SHARP/Hudson

 23536 Bytes free

Ok
MON
*L
Found  "B62218(1Z006).    "
```

画面2　ロード終了直後の画面

```
MZ-1Z006 ノ テーフ゜ ヲ セットシテ クダ゛サイ
```

画面3　MZ-1Z006のテープをロードし終わったところ

```
Found:MZ-1Z006
Found:M.L.1Z001M
Found:MZ-1Z001M
Found:M.L.2Z-004A
Found:2Z-004A
コンハ゛ータ テーフ゜ヲ モウイチト゛ セットシテクダ゛サイ
カセットノ フタ ヲ キチントシメタラ スヘ゜ースキー ヲ オシテクダ゛サイ
■
```

画面4　セーブが終わったときの画面

```
コレカラ セーフ゛ ヲ ハシ゛メマス。
キーホ゛ート゛ ニハ セ゛ッタイニ テ ヲ フレナイテ゛ クダ゛サイ。


セーフ゛ ハ フ゛シ゛ シュウリョア シマシタ。
コレカラハ IPLカラ チョクセツ ロート゛ テ゛キマス。
(CZ-8CB01 ノ カワリニ ツカエマス。)
アトテ゛ カセット ノ ツメヲ オイテ クダ゛サイネ!
ナニカ キーヲ オスカ,テ゛ンケ゛ン ヲ イレナオシテ クダ゛サイ。■
```

そっくりそのまま利用できます。たったこれだけの作業で，X1でマシン語の世界が広がるのですから素晴らしいことです。

MZ－1Z005を使うには

MZ－1Z005は，本格的なマシン語開発ツールです。

これを使うと，数十バイトから50Kバイト以上に渡る長大なプログラムまで，アセンブリ言語を使って，自由にプログラムを作成することができます。

MZ-1Z005は，5本のプログラムから成っていて，それぞれが別々のテープに入っています（ひとつだけは他のプログラムのあとに入っているため，実際は4本組になっている）。

システムソフトウェアコンバータは，B6-2218のA面を使います。（B面と，両方とも使うなら，MZ-1Z005＋MZ-1Z006が32,000円なので，約$\frac{1}{2}$の費用ですみ，かなり割安のようです。

コンバートの手順は，ほぼMZ-1Z006の場合と同じですが，テープを入れ換える（4本のカセットに分かれて入っているからです）作業が追加されるだけですので，ここでは画面を含めた説明は割愛させていただきます。

なお，MZ-1Z005についても9月号95ページに紹介記事がありますから，よかったらご覧になってください。

EA-MZを使うには

このソフトは，M(C)Z-1Z005よりも簡単なマシン語開発ツールです。

扱い方は，やはりM(C)Z-1Z005よりも簡単で，価格も安いのですが，プログラムは4～6Kバイトが限度のようで，あまり大きなプログラムは作れません。BASICで慣れたスクリーンエディットも利用できない（M(C)Z-1Z005はできる）ものの，初心者に適した部分も多いようです。

このソフトを使うには，初めに触れたとおり，CZ-1Z001かCZ-1Z002が必要です。（逆に，これらを既に使っている人にとっては，もっとも安上がりです）。CZ-1Z001などは，MZのBASICであるMZ-1Z001やMZ-1Z002を，B6-2213というシステムソフトウェアコンバータを使ってさきほどと同じ要領でコンバートしたもので，それぞれ43500バイト／39148バイトのフリーエリアを持っているものです。B6-2213は，発売元のBASIC HOUSEでもベストセラーといっているそうですので，持っている人も多いと思いますが，まだの人は購入してください（MZ-1Z001は2,150円ですので，合わせて購入してください）。

画面5 コンバード後，IPLから起動したところ

```
** Machine language CZ-1Z006 V1.0A **
           Select monitor
0)CZ-1Z001M 1)22-004A M)onitor B)oot  ?▒
```

画面6 EA-MZをロード中

```
** MONITOR CZ-1Z001M **
────────────────────────────────
  BASIC interpreter  CZ-1Z001 V1.0A
  Copyright 1982 by SHARP Corp.
────────────────────────────────
 43500 Bytes
Ready
MON
*L
FILE NAME:
LOADING EA-80B
```

画面7 EA-MZがスタートしたところ

```
*** EA-80B (Editor Assembler for MZ-80B) Ver 1.0 ***
  Copyright 1981 (C) ASCII CONSUMER PRODUCTS INC.

LEFT:30210  USED:0
>▒
```

そして，画面6のように，MON␣L␣として，モニタからロードします（ファイルネームはEA-MZではなく，旧商標のEA-80Bになっています）。やがてMZとまったく同じように，オートスタートします（画面7）。

なお，CZ-1Z001などにとってこのソフトは，他のBASICアプリケーションソフト同様に「手直しなしで使えるソフト」のひとつですが，プリンタを持っている人だけは，画面8のようにパッチする（内容を変更する）とよいでしょう。

サンプルプログラムの作成

「マシン語を扱う方法」としてはここまで説明すれば充分で，「あとは，各ソフトの取り扱い説明書と本誌8月号・9月号をご覧ください」なんてサラッと書いておいてもよいのですが，サンプルプログラムがあったほうがわかりやすいと思いましたので，何を作ったら……と考えに考え抜いた(大げさかナ!?)結果，発表したいと思います。

せっかくですから，便利で後々まで使えるものを，ということでユーティリティにしてみました。

それは，2700ボーでセーブするプログラムです。これは，CZ-1Z001，CZ-1Z002，CZ-1Z005，CZ-1Z006などのモニタのもとで利用できます。CZ-1Z00xは（既に御存知の方も多いでしょうけど），2000ボーのMZフォーマットでLOAD／SAVEされているのです。ということは，これらで作ったプログラム（マシン語プログラム）は，MZのモードでしか使えないわけです。最初に「MZとしてマシン語を使える」といった以上，これはこれでよいのですが，せっかくですから「CZ-8CB01で使えたら」というわけなのです。

実行方法はあとで説明しましょう。

入力方法

「MZ-1Z006を使うには」の項目にしたがって，3つのうちのいずれかをロードし

てください。

CZ-1Z006（マシンランゲージ）では，モニタは2Z-004Aにしておくとよいでしょう。Loading addressは，↵のみにしておいてください。そして，

[W] [D][0][0][0]

として，D000Hからダンプリスト1を入力してください。その後，

[1][1] [0][2]…

のように，16進数の部分だけ入力すればOKです。モニタのように↵を入力する必要がないため，ここだけ見ても原理上⅔の手間で打ち込めることになります。もし，SやZのような「16進数とは関係のないキー」が押されても，ピロピロッと警告音で知らせてくれ，入力されません。また，X1用にアレンジしたCZ-1Z006独自（つまりMZ-1Z006にはない）の機能として，16進テンキーモードがあります。図1を見てください。[CTRL]+[T]と押すだけ（[CTRL]キーを押しながら[T]（Ten keyの意）を押すということ。モニタCZ-1Z001Mを選択した場合はスクリーンエディット入力時に行なってください。コマンド待ち時に[B] [CTRL]+[T] ↵のようにするとよいでしょう）で，テンキー部の[HOME]が[A]に，[/]が[B]に…変身できます。これなら少し慣れれば左手でOh!MZを持ち，目で追いつつ右手でテンキーを打つ，なんてこともラクラクできるようになるでしょう。このモードは再び[CTRL]+[T]を入力することで抜けられます（要するに元に戻る）。

もし，間違った16進数を入力してしまったときも，[⇦]だけで直前のアドレスに戻れます。入力が終わったなら，よく確かめて

S↵SAVE2700bps.OBJ↵ D000 D222 00B1↵としてセーブしておいてください。

次に，CZ-1Z005（システムプログラムという名のソフト）のほうですが，起動直後に

*▒

となっていますから，[I]ORG D000↵;↵ ST:EQU 00B1H ;Monitor HOT START↵;↵START:……と，行番号をつけずにリスト1を打ち込んでください。多少のミスは後で直すことにしておいて，取りあえずは一度に全部入力してください。

最後の END ↵まで終わったら，[BREAK]を押してください。*と出てきますから，T↵としてリストをとってみてください。行番号が付いて出ますからリスト1と照合してください。

スペースキーか[ESC]で一時停止，[BREAK]で中断ができます。[TAB]を押しているとその間のスクロールアップが遅くなり，ゆっくりと見ることができます。

もしミスが発見できたら，すかさず[BREAK]して，BASICの要領でスクリーンエディットしてください。

たとえば23行目を消したい，というようなことが起きたら，

23↵

なんてせずに，* と出ているときに

23J↵1K

としてください。

41行と42行の間に何か入力したいときは，*と出ているときに

42J↵I○○………………………↵

としてください。○○…のところは入れたい内容を書きます。何行も追加したいときは続けて入力してください。これでよい，となったら[BREAK]してください。

全部入力できたら

X↵

でアセンブラに移り

[N] [A] [N] ↵

とします。これでアセンブルリストが見られます。

1192D2
CD5BD1
CD64D1
1A
⋮

と出る部分（左のほう）の附近に，P以外の表示（E，Q，V，Lなど）が出てきたときは入力ミスがあったと思われますので，[BREAK]して修正してください。

これがOKなら，[BREAK]して，エディタに戻り新しいテープをセットして

W SAVE2700bps.ASC ↵としてください。ソースリストがテープにセーブできます。

終わったら

X↵

画面8 プリンタ使用時のパッチ方法

```
** MONITOR CZ-1Z001M **
────────────────────────────────────
 BASIC interpreter  CZ-1Z002 V1.0B     } もちろんCZ-1Z001
 Copyright 1982 by SHARP Corp.
────────────────────────────────────
 39148 Bytes
Ready
MON                     MON↵
*M                      M↵
M-ADR.$01F0             01F0↵    ) オートスタート禁止
01F0 38 18              18↵
01F1 CB                 [BREAK]
*L                      L↵
FILE NAME:              ↵
LOADING EA-80B
*M                      M↵
M-ADR.$E002             E002↵    ) 1ページ長の変更
E002 32 3F              3F↵        (MZでも行なうと良い)
E003 21 -               -↵
M-ADR.$E075             E075↵    )
E075 F5 FE              FE↵
E076 CD 0D              0D↵
E077 88 C4              C4↵
E078 E0 74              74↵        パッチ
E079 CD 0E              0E↵
E07A DE DA              DA↵
E07B E0 B9              B9↵
E07C F1 E0              E0↵
E07D FE -               -↵       )
M-ADR.$E125             E125↵
E125 0D 00              00↵
E126 C8                 [BREAK]
*J                      J↵
J-ADR.$01B5             01B5↵    ) コピー（直前に新しいテープを
WRITING EA-80B                     セットすること）
*▒
```

この部分だけ打ち込んでください。

で再びアセンブラに入り，

[G] [N] [N]

SAVE2700bps，RB↵

として，リロケータブルバイナリをテープにセーブします。

次に，一度電源を切るなどして，リンカを起動します（リンカは，B6-2218のA面の先頭から[SHIFT]+[▷▷] [SHIFT]+[▷▷]とすると簡単にサーチできます）。

起動直後にE0のようにと答えると，コマンド待ちに入れますから

L D000 D000

↵

として、リロケータブルバイナリをロードします。

ロードが終わったら，間もなく目的のマシン語ができ上がっていますから，

S

SAVE2700bps.OBJ↵

D000 D222

D000 00B1

として，でき上がったプログラムをセーブしておきましょう。

最後に，EA-MZを使う場合ですが，コマンド待ち（ロード直後）に，

I↵ORG D000H↵; ↵…

として入力します。CZ-1Z005と異なり，Iに続けて1行目は入力しません。後は同様に行番号なしで入力すればよいのですが，最後の[BREAK]まで終わったらA↵でリストをとります（Tではない）。0からの行番号が付くので，リスト1とは1行ずつずれますが，よく確かめてください。

22行目を直したいなら，コマンド待ち（>と出ます）のときに。

E22↵

としてください。ここで新たに入力します（カーソルを1行上に移動して，エディットしてもよい。ただし1度には1行しかエディットできません）。

23行目を消すには

T23↵D1↵

41行目と42行目の間に入力したいなら，

T42↵I↵○○…↵

です。CZ-1Z005とコマンドの名前は違うのですが，考え方は同じなのです。

ソースリストは

S↵

SAVE2700bps.ASC↵

でセーブできます。

アセンブラには，

J↵

で入れます。

すぐにエラーの行だけリストされますから，あったときは[BREAK]して直してください。左端に出るのは行番号ですから，それを参考にするとよいでしょう。

エラーがなかったら，

O↵

としてください。これですぐにマシン語ができあがります。

[BREAK]して，；↵でモニタ（BASICのモニタ，CZ-1Z001M）に入り，

S↵

SAVE2700bps.OBJ

D000↵

D222↵

↵

としてセーブしてください。

一見CZ-1Z005よりも簡単ですが，ソースリストがメモリによって制限されてし

リスト1 マシンランゲージによるダンプリスト

```
M D000 D222
D000 11 02 D2 CD 5B D1 CD 64 D1 1A FE 0B CA B1 00 21
D010 17 00 19 11 E3 D1 01 10 00 ED B0 EB 06 11 3E 0D
D020 36 20 2B BE 20 02 10 F8 CD 7C D1 22 F6 D1 E5 CD
D030 85 D1 C1 B7 ED 42 38 F0 23 22 F4 D1 CD A9 D1 21
D040 00 00 28 05 CD A0 D1 38 F3 22 F8 D1 11 FA D1 3E
D050 ED CD BD D1 3E EF CD BD D1 1B AF 12 13 12 13 12
D060 16 01 CD 8E D1 DA B1 00 11 1A D2 CD 5B D1 21 E3
D070 D1 06 10 7E 23 CD 73 D1 10 F9 F3 01 40 1F CD BD
D080 D0 01 28 04 D4 CC D0 21 E2 D1 01 20 00 D4 EE D0
D090 11 20 03 3E E6 D4 3F D1 38 1D 1B 7A B3 20 F4 01
D0A0 14 0C CD CC D0 2A F6 D1 ED 4B F4 D1 D4 EE D0 38
D0B0 06 01 E8 03 CD BD D0 CD 97 D1 C3 B1 00 CD 4F D1
D0C0 D8 3E F5 3D 20 FD 0B 78 B1 20 F2 C9 69 0C 61 CD
D0D0 0A D1 CD 4F D1 D8 0B 78 B1 20 F4 D4 25 D1 D4 4F
D0E0 D1 2D 20 F7 D4 0A D1 D4 4F D1 25 20 F7 C9 D9 21
D0F0 00 00 D9 7E CD 3F D1 D8 23 0B 78 B1 20 F5 D9 7C
D100 CD 3F D1 7D D4 3F D1 D9 23 D9 F5 C5 01 03 1A ED
D110 57 AF ED 79 3E 44 CD 56 D1 3E 01 ED 79 3E 3B CD
D120 56 D1 C1 F1 C9 F5 C5 01 03 1A 3E 00 ED 79 3E 20
D130 CD 56 D1 3E 01 ED 79 3E 18 CD 56 D1 C1 F1 C9 C5
D140 06 08 CD 0A D1 07 DC 07 D1 D4 25 D1 10 F7 C1 3E
D150 1A D8 01 1F 3F C9 3D C2 56 D1 C9 CD D8 D1 CA AC
D160 06 C3 B6 05 CD D8 D1 11 00 11 CA E5 0B 11 AB 10
D170 C3 A4 06 CD D8 D1 CA 3C 06 C3 C6 08 CD D8 D1 CA
D180 3F 05 C3 7A 05 CD D8 D1 CA 42 05 C3 83 05 CD D8
D190 D1 CA 16 04 C3 57 04 CD D8 D1 CA 95 04 C3 CE 04
D1A0 CD D8 D1 CA A2 05 C3 14 06 CD D8 D1 3E 4A 20 06
D1B0 32 54 05 C3 50 05 32 99 05 C3 98 05 E9 CD D8 D1
D1C0 21 87 04 CC 9B 04 28 06 21 C3 04 CD D4 04 06 03
D1D0 CD BC D1 12 13 10 F9 C9 C5 47 3A ED 00 EE 11 78
D1E0 C1 C9 01 00 00 00 00 00 00 00 00 00 00 00 00 00
D1F0 00 00 00 20 00 00 00 00 00 00 00 00 00 00 00 00
D200 00 00 46 49 4C 45 20 4E 41 4D 45 20 66 6F 72 20   ..FILE NAME for
D210 43 5A 2D 38 43 42 30 31 3A 0D 57 72 69 74 69 6E   CZ-8CB01:.Writin
D220 67 3A 0D                                          g:.
```

まいますので，あまり大きなプログラムは作れないわけです。

ＣＺ-１Ｚ005なら，ソースリストを10分割しておくようなこともできるため，数十Ｋバイトのプログラムも作成できます。一般に，ソースリストは目的のマシン語の10倍程度の大きさになりますから，EA-MZでは無理が出てくる，というわけです。

なお，ＯＲＧ　Ｄ０００Ｈ　をもっと別の値にすれば，好きなアドレスにマシン語が作れます。例えば　ＯＲＧ　７０００Ｈに変えれば，７０００Ｈ以降に作られます。それに，このプログラムを改造する，というのもソースリストがあれば簡単にできます（個人的使用にとどめてください）。

使い方

このプログラムは，モニタ（CZ-1Ｚ001Ｍと2Ｚ-004Ａがありますが，どちらでもＯＫです）のもとで動きます。

Ｌ⏎⏎

でロードしたら，

Ｊ⏎

Ｄ０００⏎

で実行できます。

このときに，

ＦＩＬＥ　ＮＡＭＥ　for　ＣＺ-８Ｃ

Ｂ０１：

と聞いてきますから，ファイルネームを入力してください。

続いて

Ｓ-ＡＤＲ．＄

と

Ｅ-ＡＤＲ．＄

と

Ｊ-ＡＤＲ．＄

とを聞いてきますので，スタート・エンド・実行開始アドレスを順に入力してください。これはモニタのＳコマンドとほとんど同じなのですが，2700ボーでＳＡＶＥされている点だけが異なります。

ＣＺ-8ＣＢ０１を起動したら，モニタでロードしてみてください。

おわりに

いささか急ぎ足になってしまいましたが，Ｘ１でもマシン語が扱える，ということがわかっていただけたかと思います。

やっとＭＺに追いついた，というと何か「Ｘ１は遅れてるな〜…」という気がしないでもありませんが，他社機と比較すれば，テープベースでこれだけ色々とできるものはほかにありません。マシン語についてもトップクラスに入れた，という表現のほうが正しいかも知れませんね。

ここで触れた３つのソフトは，いずれももっと幅の広い応用ができ，説明したのはほんの一部です。何となく難かしそうだと感じてしまった人も，実際に使ってみると案外楽なものです（１分でできることが，文字では何百字・何千字にもなってしまうものですから)。どんどん使って，どんどん腕をみがこうではありませんか。

画面９　サンプルプログラムで自分自身をセーブしてみた

```
** MONITOR CZ-1Z001M **

BASIC interpreter  CZ-1Z001 V1.0A
Copyright 1982 by SHARP Corp.

 43500 Bytes
Ready
MON
*L
FILE NAME:
LOADING SAVE2700bps.OBJ
*J
J-ADR.$D000
FILE NAME for CZ-8CB01:TEST!!
S-ADR.$D000
E-ADR.$D222
J-ADR.$
Writing:TEST!!
```

画面10　画面９で2700ボーSAVEしたものをロードしたところ。

```
SHARP-HuBASIC CZ-8CB01 V1.0
Copyright (C) 1982 by SHARP/Hudson

 23536 Bytes free

Ok
MON
*L
Found  "TEST!!         .   "
*D D000
:D000=11 02 D2 CD 5B D1 CD 64 /..メヘ[ムヘd
:D008=D1 1A FE 0B CA B1 00 21 /ム.生.ハア.!
:D010=17 00 19 11 E3 D1 01 10 /.....♥ム..
:D018=00 ED B0 EB 06 11 3E 0D /.▞ー▪..>.
:D020=36 20 2B BE 20 02 10 F8 /6 +セ ..時
:D028=CD 7C D1 22 F6 D1 E5 CD /ヘ|ム"月ム♣ヘ
:D030=85 D1 C1 B7 ED 42 38 F0 /■ムチキ▞B8▒
:D038=23 22 F4 D1 CD A9 D1 21 /#"水ムヘゥム!
:D040=00 00 28 05 CD A0 D1 38 /..(.ヘ ム8
:D048=F3 22 F8 D1 11 FA D1 3E /木"時ム.秒ム>
:D050=ED CD BD D1 3E EF CD BD /▞ヘスム>□ヘス
:D058=D1 1B AF 12 13 12 13 12 /ム.ッ.....
:D060=16 01 CD 8E D1 DA B1 00 /..ヘ■ムレア.
:D068=11 1A D2 CD 5B D1 21 E3 /..メヘ[ム!♥
:D070=D1 06 10 7E 23 CD 73 D1 /ム..ー#ヘsム
:D078=10 F9 F3 01 40 1F CD BD /.分木.@.ヘス
*■
```



リスト2　EA-MZによるアセンブルリスト

```
***  EA-80B (Editor Assembler for MZ-80B) Ver 1.0 ***     A.C.P.        PAGE     1

0000                       ORG   D000H
0001                ;
0002 00B1           ST:    EQU   00B1H          ;Monitor HOT START
0003                ;
0004 D000 1102D2    START: LD    DE,FNM
0005 D003 CD5BD1           CALL  NLMSG          ;'File name'
0006 D006 CD64D1           CALL  GETL           ;INPUT File name
0007 D009 1A               LD    A,(DE)
0008 D00A FE0B             CP    0BH            ;BREAK
0009 D00C CAB100           JP    Z,ST
0010 D00F 211700           LD    HL,23
0011 D012 19               ADD   HL,DE
0012 D013 11E3D1           LD    DE,NAME
0013 D016 011000           LD    BC,16
0014 D019 EDB0             LDIR
0015 D01B EB               EX    DE,HL
0016 D01C 0611             LD    B,17
0017 D01E 3E0D             LD    A,0DH
0018 D020 3620      FIL20H:LD    (HL),20H
0019 D022 2B               DEC   HL
0020 D023 BE               CP    (HL)
0021 D024 2002             JR    NZ,INPADR
0022 D026 10F8             DJNZ  FIL20H
0023 D028 CD7CD1    INPADR:CALL  SSET           ;'S-ADR.$'
0024 D02B 22F6D1           LD    (DTADR),HL     ;SAVE start adrs.
0025 D02E E5               PUSH  HL
0026 D02F                  CALL  ESET           ;'E-ADR.$'
```

リスト3　CZ-1Z005によるアセンブルリスト

```
   **  Z80 ASSEMBLER 1Z-005B PAGE 01  **

01 D000                        ORG     D000H
02 D000              ;
03 00B1    P         ST:       EQU     00B1H        ;Monitor HOT START
04 D000              ;
05 D000 1102D2       START:    LD      DE,FNM
06 D003 CD5BD1                 CALL    NLMSG        ;'File name'
07 D006 CD64D1                 CALL    GETL         ;INPUT File name
08 D009 1A                     LD      A,(DE)
09 D00A FE0B                   CP      0BH          ;BREAK
10 D00C CAB100                 JP      Z,ST
11 D00F 211700                 LD      HL,23
12 D012 19                     ADD     HL,DE
13 D013 11E3D1                 LD      DE,NAME
14 D016 011000                 LD      BC,16
15 D019 EDB0                   LDIR
16 D01B EB                     EX      DE,HL
17 D01C 0611                   LD      B,17
18 D01E 3E0D                   LD      A,0DH
19 D020 3620         FIL20H:   LD      (HL),20H
20 D022 2B                     DEC     HL
21 D023 BE                     CP      (HL)
22 D024 2002                   JR      NZ,INPADR
23 D026 10F8                   DJNZ    FIL20H
24 D028 CD7CD1       INPADR:   CALL    SSET         ;'S-ADR.$'
25 D02B 22F6D1                 LD      (DTADR),HL   ;SAVE start adrs.
26 D02E E5                     PUSH    HL
```

リスト4　2700ボーSAVEプログラム

```
ORG D000H
;
ST:EQU 00B1H ;Monitor HOT START
;
START:LD DE,FNM
CALL NLMSG ;'File name'
CALL GETL ;INPUT File name
LD A,(DE)
CP 0BH ;BREAK
JP Z,ST
LD HL,23
ADD HL,DE
LD DE,NAME
LD BC,16
LDIR
EX DE,HL
LD B,17
LD A,0DH
FIL20H:LD (HL),20H
DEC HL
CP (HL)
JR NZ,INPADR
DJNZ FIL20H
INPADR:CALL SSET ;'S-ADR.$'
LD (DTADR),HL ;SAVE start adrs.
PUSH HL
CALL ESET ;'E-ADR.$'
POP BC
OR A
SBC HL,BC
JR C,INPADR
INC HL
LD (SIZE),HL ;HL=Length
KINP2:CALL JINP ;'J-ADR.$'
LD HL,0
JR Z,SAVE ;only CR
CALL HLHEX
JR C,KINP2 ;input error
SAVE:LD (EXADR),HL ;Execute adrs.
LD DE,IBU24
LD A,EDH
CALL INP ;DATE(YY/DM/DD)input
LD A,EFH
CALL INP ;TIME(HH:MM:SS)input
DEC DE
XOR A
LD (DE),A ;Kill SEC.
INC DE
LD (DE),A ;Record no. L
INC DE
LD (DE),A ;Record no. H
;
LD D,1 ;Write mode
CALL MOTOR ;CMT=10
JP C,ST ;Write protect etc.
LD DE,WRIMES ;'Writing'
CALL NLMSG ;NewLine and Mesage
LD HL,NAME
LD B,16
PRLOOP:LD A,(HL) ;PRINT file name LOOP
INC HL
CALL PRNT ;Print 1 character
DJNZ PRLOOP
;
DI
LD BC,8000
CALL TSPE ;Wait 8 sec.
LD BC,1064 ;1064 bit
CALL NC,GAP
LD HL,IBUFE ;from
LD BC,0020H ;len
CALL NC,WTAPE ;SAVE
LD DE,800
WRAPS1:LD A,230 ;Wait short
CALL NC,WBYTE
JR C,STPRET ;BREAK
DEC DE ;800 times
LD A,D
OR E
JR NZ,WRAPS1
;
LD BC,3092 ;3092 bit
CALL GAP
LD HL,(DTADR) ;from
LD BC,(SIZE) ;len
CALL NC,WTAPE ;SAVE
JR C,STPRET ;BREAK
LD BC,1000
CALL TSPE ;Wait 1 sec.
STPRET:CALL MSTOP ;CMT=1
JP ST ;GO Monitor
;
TSPE:CALL ?BRK ;IF BREAK THEN exit.
RET C ;BREAK
LD A,245
TSPE1:DEC A
JR NZ,TSPE1 ;Wait 1ms.
DEC BC
LD A,B
OR C
JR NZ,TSPE
RET
;
GAP:LD L,C
INC C
LD H,C
GAP1:CALL LONG ;it sounds 'pow'
CALL ?BRK ;IF BREAK THEN exit.
RET C ;BREAK
DEC BC
LD A,B
OR C
JR NZ,GAP1 ;number of '1'
GAP2:CALL NC,SHORT ;'pee'
CALL NC,?BRK
DEC L
JR NZ,GAP2 ;a number of '0'
GAP3:CALL NC,LONG ;'pow'
CALL NC,?BRK
DEC H
JR NZ,GAP3 ;a number of '1'
RET
;
WTAPE:EXX
LD HL,0 ;Initialize Check Sum.
EXX
WTAPE1:LD A,(HL)
CALL WBYTE ;1byte write to tape
RET C ;BREAK
INC HL
DEC BC
LD A,B
OR C
JR NZ,WTAPE1 ;Repeat 'BC' times
EXX
LD A,H ;Check sum. High byte
CALL WBYTE
LD A,L ;Check sum. Low byte
```

```
CALL NC,WBYTE
;
LONGHL:EXX
INC HL ;Increment check sum.
EXX
LONG:PUSH AF ;Write bit'1' to tape
PUSH BC ;(250 ms. pulse)
LD BC,1A03H
LD A,I
XOR A
OUT (C),A ;Low
LD A,68
CALL DLY
LD A,1
OUT (C),A ;Higth
LD A,59
CALL DLY
POP BC
POP AF
RET
;
SHORT:PUSH AF ;Write bit'0' to tape
PUSH BC ;(125 ms. pulse)
LD BC,1A03H
LD A,0
OUT (C),A ;Low
LD A,32
CALL DLY
LD A,1
OUT (C),A ;High
LD A,24
CALL DLY
POP BC
POP AF
RET
;
WBYTE:PUSH BC ;Write 1 byte
LD B,8 ;8 bit
CALL LONG ;no check sum
WBYTE1:RLCA
CALL C,LONGHL ;'1' bit,chek sum
CALL NC,SHORT ;'0' bit
DJNZ WBYTE1
POP BC
;
?BRK:LD A,1AH ;Check 'BREAK'
IN A,(01H)
RRA
CCF ;Cy=1 ... BREAK
RET
;
DLY:DEC A ;Delay
JP NZ,DLY
RET
;
NLMSG:CALL ?MON
JP Z,06ACH ;2Z-004A
JP 05B6H ;CZ-1Z001M
;
GETL:CALL ?MON
LD DE,1100H
JP Z,0BE5H ;2Z-004A
LD DE,10ABH
JP 06A4H ;CZ-1Z001M
;
PRNT:CALL ?MON
JP Z,063CH ;2Z-004A
JP 08C6H ;CZ-1Z001M
;
SSET:CALL ?MON
JP Z,053FH ;2Z-004A
JP 057AH ;CZ-1Z001M
;
ESET:CALL ?MON
JP Z,0542H ;2Z-004A
JP 0583H ;CZ-1Z001M
;
MOTOR:CALL ?MON
JP Z,0416H ;2Z-004A
JP 0457H ;CZ-1Z001M
;
MSTOP:CALL ?MON
JP Z,0495H ;2Z-004A
JP 04CEH ;CZ-1Z001M
;
HLHEX:CALL ?MON
JP Z,05A2H ;2Z-004A
JP 0614H ;CZ-1Z001M
;
JINP:CALL ?MON
LD A,4AH ;J
JR NZ,JINP1
LD (0554H),A
JP 0550H
JINP1:LD (0599H),A
JP 0598H
;
JPHL:JP (HL)
;
INP:CALL ?MON
LD HL,0487H
CALL Z,049BH ;2Z-004A Command out
JR Z,INP1 ;2Z-004A
LD HL,04C3H ;CZ-1Z001M
CALL 04D4H ;Command out to SUB.
INP1:LD B,3
INP2:CALL JPHL ;Get data from SUB.CPU
LD (DE),A ;Input data
INC DE
DJNZ INP2
RET
;
?MON:PUSH BC ;IF 2Z-004A THEN Z=1
LD B,A
LD A,(00EDH)
XOR 11H
LD A,B
POP BC
RET
;
IBUFE:DEFB 01H ;File type
NAME:DEFS 16 ;File name
DEFB 20H ;Pass word
SIZE:DEFS 2 ;File size
DTADR:DEFS 2 ;Start adrs
EXADR:DEFS 2 ;Exec. adrs
IBU24:DEFS 3 ;Date
DEFS 2 ;Time
DEFS 3 ;for Disk
;
FNM:DEFM 'FILE NAME for CZ-8CB01:'
DEFB 0DH
;
WRIMES:DEFM 'Writing:'
DEFB 0DH
;
KBUF:DEFS 81 ;Filename input buffer
;
END
```

X1 (HuBASIC)
THE PRO-BOWLING

act.1 そしてピンが倒れた アニメーショングラフィックス

Elder Man

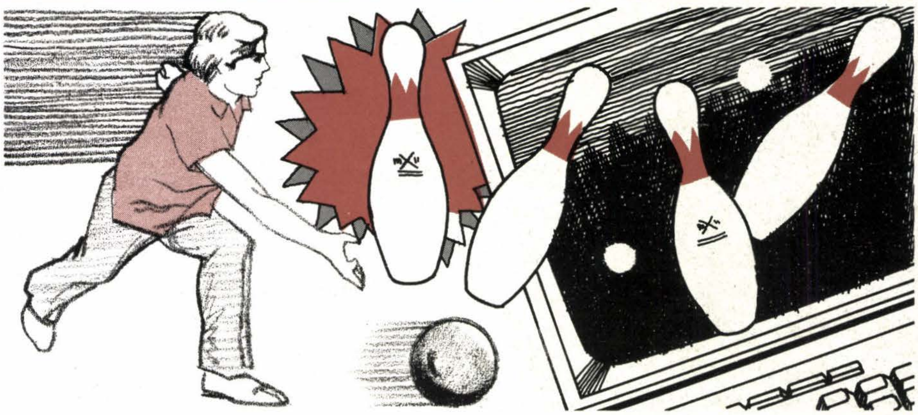
先日,Oh！MZの編集室に立ち寄りました。私が訪れるときは，いつも忙しそうです。月末に行くことが多かったので，MZ 誌の編集のツメの段階に入っているためでしょうか。

編集室に行ったときは，いつもアンケートはがきをみせてもらうことにしています。ちょうど，ゲームプログラミング入門の3回の連載を終えたところだったので，その評価に興味津々といったところでした。ゲームプログラムの紹介だったら，苦にもならないのですが，プログラミングの記事でしかも連載というのは大変しんどいものだったので，評判が気にならないといったら嘘になります。手きびしい批判もありましたねえ……「もっとおもしろくして」とか「ゲームが幼稚」だとか……。しかし，X１のユーザーから多く好評をいただいたため，実際，もうこりごりと思っていたところなのですが，ゲームプログラミングの連載に再度挑戦することになりました。

前回の連載では，X１のBASICプログラム入門ということで，比較的やさしいゲームプログラムを作成しました。今回は，市販品としてもおかしくないレベルのゲームを実際に作るという過程を紹介します。前回を初級コースとするなら今回は中級コースというところでしょうか。

●ゲームのアイデア

今回のゲームの素材はボーリングです。ELDER MANのゲームソフトの制作のうえでの信条のひとつに，スポーツはゲームソフトにしないというのがあります。それは，実際にそのスポーツをやることのほうがおもしろいという理由からです。ボーリングもスポーツだと思いますし，また既にボーリングのゲームも発売されていますが，これから作るゲームは，ボーリングそのも



のを楽しむということだけではないアイデアを入れて完成させたいと思っています。そのアイデアについては，最終回まで待ってください。今回と次回では，とにかくボーリング自身をプログラミングします。実際にあるスポーツをプログラムするということは，プログラミングの技術として大いに役に立つことと思いますので，ELDER MAN流プログラミング技法におつき合いください。

●ゲーム作りのプラン

私の場合，ゲームのプログラミングに入る前に頭の中でゲームの基本的なアイデアをまとめ，さらにプログラムのディテールについてどのように作るかを考えます。すべてがOKと判断されたあと，ようやくX１のキーボードに向うことになります。ですから，キーボードをたたいているときはプログラムの方法について考えているというより，プログラミングをしていることになります。よく，１日中マイコンに向かっているのかと，いろいろな人から聞かれますが，どちらかというと考えているだけの時間のほうが長いのです。ゲームの一番のポイントはアイデアなので，おもしろいアイデアが浮かばないと，いつまでもただ考え続けるだけの時を過ごすことになります。

ともあれ，アイデアがまとまり制作に入ることになると，実際にプログラムを作る手順はだいたい，いつも次のようになります。

① ゲームで使う絵を描くツールソフトを用意する（自分で作ったり，買ってきたりします)。
② ゲームで使う絵を描くプログラムを作る。
③ メインプログラムで描く，画面のレイアウトを決める。
④ 各種処理サブルーチンを作る。
⑤ メインルーチンを作りテストランを行なう。
⑥ バグ取りや，アイデアの追加，変更を行なう。
⑦ MUSICを入れる。
⑧ ゲームの説明やデモ用のプログラムを作る。
⑨ もう一度全体を見直して，仕上げる。

この中で，一番時間のかかるのが②です。カラーで1ドットごとに作るため，とにかくたいへんで，何とか楽に作るためにはPCG EDITORなどのツールソフトを利用する必要があります。

●PCG EDITORの作成

ボーリングのピンが倒れるところを描くのには，PCG (プログラマブル　キャラク

タ　ジェネレータ）を利用します。英数字などと同程度の速さで，しかも，ドットごとの色指定ができるので，ゲーム作りには大変役に立つものです。私のX1購入の動機のひとつにこのPCGがありました。しかし，この絶大な表現力を有するPCGが標準装備されているのはよいのですが，絵の定義はたいへんめんどうなことになっています。X1付属のBASICマニュアルで説明されていますが，グラフ用紙に書いて2進数を求め，16進数に変換するなどとやっていては10コのキャラクタを定義するだけでうんざりしてしまいます。

X1のCRTを利用して絵を自由に描き，描き終わるとその絵のデータが自動的にプログラムされるといったことができれば，かなり楽になります。このようなことをやってのけるのがPCG EDITORと呼ばれるツール（道具の意味）ソフトなのです。PCGで作る絵も，ゲームごとにその大きさが変化するので，ゲームを作る前に一番使いやすいようにしておくと便利です。16×16ドットつまりキャラクタ4コ分を同時に定義できるPCG EDITORがあっても，24×24ドットの絵をそこから作るのは楽ではありませんから。

今回のボーリングゲームでは，ピンの数は，正面からみるとかくれる部分を除いて7本に見えるので，ピン1本をひとつのキャラクタの大きさに割りあて，横に7コのキャラクタをつなげて全部のピンを描きます。1列では背が低いので，上下2コのキャラクタをつなげて，7×2＝14コのキャラクタの集まりで描くことにします。画面の表示文字数について書くのを忘れていましたが，同一画面上で得点を表示するため，横80文字モードにする必要があります。ですから，7×2コのキャラクタといっても，横方向に圧縮されるので細長すぎることはありません。この14コのキャラクタ分にまとめて絵を描けるようなPCG EDITORがあると作業がやりやすくなります。

リスト1の後半に，このために改造したPCG EDITORを載せてあります。50000行からのプログラムですが，RUN 50000 ⏎として絵を定義すると，それ以前の行番号がついてPCGの定義データがDATA文として自動的に作成されるようになっています。リスト1はこのように作成されたものです。プログラムが完成したら，PCG EDITORはプログラムから消去してしまいます。PCGは，一度定義すると再定義しなければ消えないのでこのボーリングゲームは，絵を定義するプログラムとメインプログラムの2本で構成します（PCGのデータを含めて1本のプログラムにすると23Kしかないメモリがパンクするので）。

●PCG　EDITORの使い方

RUN 50000 ⏎でプログラムスタートします。7×2（56×16ドット）のキャラクタ分のます目が描かれます。点滅するカーソルをテンキーを使って移動させ，5のキーでドットをセットし，次にCOLOR番号（0～7）を入力します（0はリセット）。絵を描き終えたら［G］のキーを入力し，INPUT CHR$ No.と聞いてくるので，1～18までの数を入力してください。14コのキャラクタの絵を18コまで定義できるということです。たとえば1を入力すると，DATA文は10000行から作成され，CRT上に表示されていきます。14行分の表示が終わったらカーソルを一番上の行に移動させて，順々に⏎キーを押します。これでプログラムとして登録されます。

一度定義した絵といっても修正作業は必要です。その場合は，RUN 50000とした後に［R］キーを入力するとINPUT CHR$ No.と聞いてくるので，呼びだしたいNo.を入力してください。画面の右上に実物大の絵が描かれます。修正する場合，さらに［Y］キーを入力すると，画面のます目に定義したときと同じ絵が再現されて，再び定義をやり直すことができます。カーソルの点滅，移動の速さは50030行のTEMPO文で変えられます。

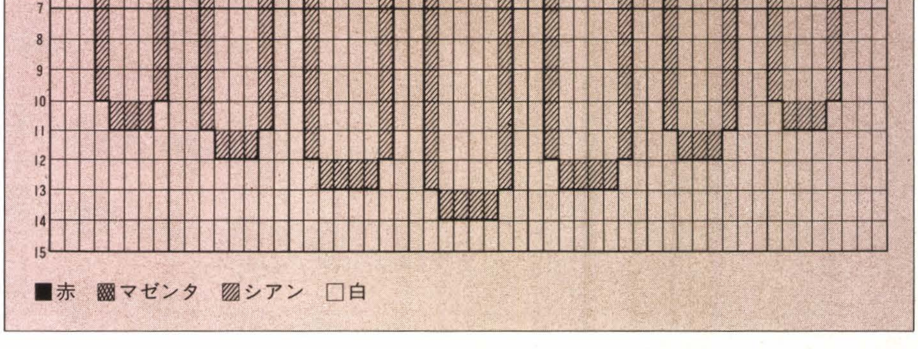
●ピンを描く

第1投目にセットされる10本のピンは図1のように描きます。ヘッドピンからだんだん小さくなるように描いてあります。この絵のデータはリスト1の10000行～10130行です。10200行～10330行ではあとにかくれているピンを描いています。次回になりますが，この絵は，第1投目の残りピンとして，第2投目で使用するものとして用意してあります。

ボーリングは何といってもストライクを出すことが一番気持ちのいいものです。ストライクについては，PCGを駆使して演出しなければゲームがおもしろくありません。

図2に示したように6コマのアニメーションでストライクを表現します。何と，これで84コ分のキャラクタを使ってしまいます。ストライクはポケットヒットだけでなく，左側から入るブルックリンと呼ばれるものもあります。もうひとつのストライクパターンを用意しなければなりません。ここで少し手抜きをして，今のポケットヒットのストライクパターンを左右反転させたPCGデータを作ります。プログラム1のPCG EDITORの58010行のCG$＝CGPAT$(I)の前にCG$＝MIRROR$(CG$)を入れて左右反転パターンを作ってあります（ただし作成されるDATA文の行番号は修正する必要があります）。

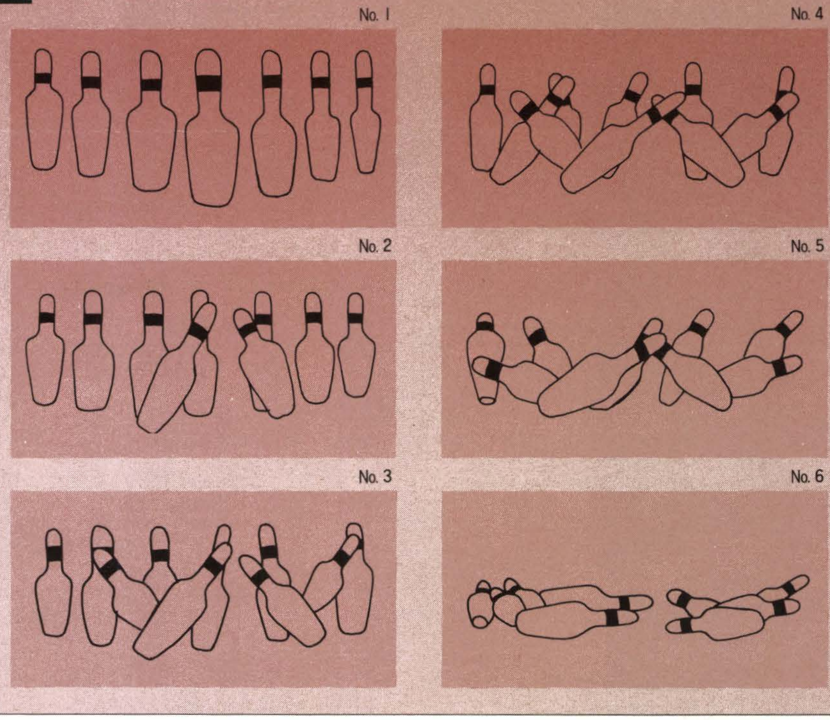
図1　ピンのレイアウト



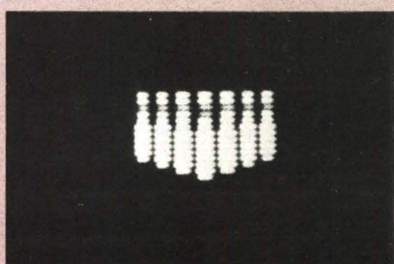
これらのストライクパターンのDATA文が，12330行までです。

本当はストライクだけでなく，すべてのケースについてピンアクションをアニメ風に表現できればいいのですが，ピンの残り方はたくさんのケースがあるため，とてもメモリが足りません。そこで，ストライクの他に1本だけ残るケース（ピンの番号でいえば，⑤，⑦，⑩）についてはストライクと同じようにアニメ風に表現することにしました。他のピンが倒れても，そのピンだけ残るようにして使用するためのものが残りの行番号です。ただし，後半にはストライクの演出をするための王冠マークが入っています。プログラム1をRUNさせるとストライクや1本残りのピンアクションを順々にみることができます。このPCGで作られる絵は最後まで修正しない予定ですから，キー入力される方は少々長いですけど頑張ってください。

図2　ストライクの表示



リスト1



```
10 '============================================================
20 '
30 ' BOWLING (DEF PCG PRO.)  Copyright  1983  by  ELDER MAN  Vr.1.0
40 '
50 '============================================================
60 WIDTH 40:DEFINTA-Z:COLOR5:CLICKOFF
70 PRINTSTRING$(40,"="):PRINT:PRINT:COLOR7
80 PRINT"   BOWLING (DEF PCG)  by  ELDER  MAN"
90 PRINT:PRINT:PRINT"                 with   PCG EDITOR (7*2)"
100 COLOR5:PRINT:PRINT:PRINT:PRINTSTRING$(40,"=")
110 COLOR3:LOCATE10,16:PRINT "WAIT A MOMENT !":BEEP:COLOR7
4997 '
4998 ' ====== DATA READ & DEF PCG ======
4999 '
5000 RESTORE 10000
5010 FOR I=35 TO 244:READ A$:DEFCHR$(I)=HEXCHR$(A$):NEXT
5017 '
5018 ' ====== PCG TEST DISPLAY ======
5019 '
5020 WIDTH 80:COLOR7
5027 '
5028 ' ====== POCKET HIT ======
5029 '
5030 FOR MD=1 TO 4:GOSUB6000:NEXT
5037 '
5038 ' ====== BROOKLYN ======
5039 '
5040 FOR MD=1 TO 4:GOSUB7000:NEXT
5047 '
5048 ' ====== STRIKE ! ======
5049 '
5050 CGEN1:FOR I=1 TO 5:COLOR2:LOCATE36,6:PRINT"▪▄▀":LOCATE36,7:PRINT"金木木":BEEP
5060 COLOR7:LOCATE36,6:PRINT"▪▄▀":LOCATE36,7:PRINT"金木木":BEEP:NEXT:CGEN
5070 LOCATE36,6:PRINT"   ":LOCATE36,7:PRINT"   ":GOTO 5030
5997 '
5998 ' ====== DISPLAY SUBROUTINE ======
5999 '
6000 CGEN1:LOCATE34,10:PRINT"#$%&'()":LOCATE34,11:PRINT"*+,-./0"
6010 PAUSE10:BEEP:LOCATE34,10:PRINT"?@ABCDE":LOCATE34,11:PRINT"FGHIJKL"
6020 PAUSE1:IF MD<4 THEN LOCATE34,10:PRINT"MNOPQRS":LOCATE34,11:PRINT"TUVWXYZ" E
LSE LOCATE34,10:PRINT"MNO°QRS":LOCATE34,11:PRINT"TUV◢XYZ"
6030 PAUSE1:ON MD GOSUB 6700,6700,6800,6900
6040 PAUSE1:ON MD GOSUB 6100,6200,6300,6920
```

```
6050 PAUSE1:ON MD GOSUB 6400,6500,6600,6940
6060 CGEN:PAUSE5:RETURN
6100 LOCATE34,10:PRINT"ijklmno":LOCATE34,11:PRINT"pqrstuv":RETURN
6200 LOCATE34,10:PRINT"ヒjklmno":LOCATE34,11:PRINT"メqrstuv":RETURN
6300 LOCATE34,10:PRINT"ijklmnホ":LOCATE34,11:PRINT"pqrstuユ":RETURN
6400 LOCATE34,10:PRINT"wxyz{|}":LOCATE34,11:PRINT"￣π_▁▂▃":RETURN
6500 LOCATE34,10:PRINT"フxyz{|}":LOCATE34,11:PRINT"モπ_▁▂▃":RETURN
6600 LOCATE34,10:PRINT"wxyz{|マ":LOCATE34,11:PRINT"￣π_▁▂ヨ":RETURN
6700 LOCATE34,10:PRINT"[¥]^_`a":LOCATE34,11:PRINT"bcdefgh":RETURN
6800 LOCATE34,10:PRINT"[¥]^_`ヘ":LOCATE34,11:PRINT"bcdefgト":RETURN
6900 LOCATE34,10:PRINT"[¥]ミ_`a":LOCATE34,11:PRINT"bcdラfgh":RETURN
6920 LOCATE34,10:PRINT"ijkムmno":LOCATE34,11:PRINT"pqrリtuv":RETURN
6940 LOCATE34,10:PRINT"wxyル{|}":LOCATE34,11:PRINT"￣π_●▂▃":RETURN
7000 CGEN1:LOCATE34,10:PRINT"#$%&'()":LOCATE34,11:PRINT"*+,-./0"
7010 PAUSE10:BEEP:LOCATE34,10:PRINT"▄▆█ ▏▍▋":LOCATE34,11:PRINT"▊██/─|┴"
7020 PAUSE1:IF MD<4 THEN LOCATE34,10:PRINT"┤ ├┐┘└":LOCATE34,11:PRINT"┌ ◡ ◠ " E
LSE LOCATE34,10:PRINT"┤ ◣┐┘└":LOCATE34,11:PRINT"┌ ▚◠"
7030 PAUSE1:ON MD GOSUB 7100,7400,7100,7900
7040 PAUSE1:ON MD GOSUB 7200,7500,7600,7920
7050 PAUSE1:ON MD GOSUB 7300,7700,7800,7940
7060 CGEN:PAUSE5:RETURN
7100 LOCATE34,10:PRINT"。「」、・ヲァ":LOCATE34,11:PRINT"ィゥェォャュョ":RETURN
7200 LOCATE34,10:PRINT"ッーアイウエオ":LOCATE34,11:PRINT"カキクケコサシ":RETURN
7300 LOCATE34,10:PRINT"スセソタチツテ":LOCATE34,11:PRINT"トナニヌネノハ":RETURN
7400 LOCATE34,10:PRINT"レ「」、・ヲァ":LOCATE34,11:PRINT"ロゥェォャュョ":RETURN
7500 LOCATE34,10:PRINT"ローアイウエオ":LOCATE34,11:PRINT"♠キクケコサシ":RETURN
7600 LOCATE34,10:PRINT"ッーアイウエン":LOCATE34,11:PRINT"カキクケコサ◆":RETURN
7700 LOCATE34,10:PRINT"フセソタチツテ":LOCATE34,11:PRINT"♥ナニヌネノハ":RETURN
7800 LOCATE34,10:PRINT"スセソタチッ゛":LOCATE34,11:PRINT"トナニヌネノ♣":RETURN
7900 LOCATE34,10:PRINT"。「」X・ヲァ":LOCATE34,11:PRINT"ィゥェ□ャュョ":RETURN
7920 LOCATE34,10:PRINT"ッーア▘ウエオ":LOCATE34,11:PRINT"カキク▒コサシ":RETURN
7940 LOCATE34,10:PRINT"スセソ▖チツテ":LOCATE34,11:PRINT"トナニ土ネノハ":RETURN
9997 '
9998 ' ====== PCG DATA ======
9999 '
10000 DATA 000E0E000E0E1F1F0004040E04040E0E000E0E000E0E1F1F
10010 DATA 001C1C001C1C3E3E0008081C1C081C1C001C1C00001C3E3E
10020 DATA 003C3C003C3C3C7E0018183C3C18183C003C3C00003C3C7E
10030 DATA 003C3C00003C3C7E0018183C3C18183C003C3C00003C3C7E
10040 DATA 003C3C003C3C3C7E0018183C3C18183C003C3C00003C3C7E
10050 DATA 0038380038387C7C0010103838103838003838000038 7C7C
10060 DATA 007070007070F8F800202070702020707000707000707 0F8F8
10070 DATA 1F1F1F0E00000000E0E0E0000000001F1F1F0E00000000
10080 DATA 3E3E3E3E1C0000001C1C1C1C000000003E3E3E3E1C000000
10090 DATA 7E7E7E7E7E3C00003C3C3C3C3C0000007E7E7E7E7E3C0000
10100 DATA 7E7E7E7E7E7E3C003C3C3C3C3C3C00007E7E7E7E7E7E3C00
10110 DATA 7E7E7E7E7E3C00003C3C3C3C3C0000007E7E7E7E7E3C0000
10120 DATA 7C7C7C7C38000000383838380000007C7C7C7C38000000
10130 DATA F8F8F8700000000070707000000000F8F8F87000000000
10200 DATA 000E0E000E0E1F1F0004040E04040E0E000E0E000E0E1F1F
10210 DATA 001C1C001C1C3E3E0008081C1C081C1C001C1C00001C3E3E
10220 DATA 0038380038387C7C0010103810103838003838003838 7C7C
10230 DATA 0038380038387C7C0010103838103838003838000038 7C7C
10240 DATA 003838003838387C0010103810101038003838003838387C
10250 DATA 0038380038387C7C0010103838103838003838000038 7C7C
10260 DATA 007070007070F8F800202070702020707000707000707 0F8F8
10270 DATA 1F1F1F0E00000000E0E0E0000000001F1F1F0E00000000
10280 DATA 3E3E3E3E1C0000001C1C1C1C000000003E3E3E3E1C000000
10290 DATA 7C7C7C38000000003838380000000007C7C7C3800000000
10300 DATA 7C7C7C7C38000000383838380000007C7C7C7C38000000
10310 DATA 7C7C7C38000000003838380000000007C7C7C3800000000
10320 DATA 7C7C7C7C38000000383838380000007C7C7C7C38000000
10330 DATA F8F8F8700000000070707000000000F8F8F87000000000
10400 DATA 000E0E000E0E1F1F0004040E04040E0E000E0E000E0E1F1F
10410 DATA 001C1C001C1C3E3E0008081C1C081C1C001C1C00001C3E3E
10420 DATA 003C3C003C3C3C7F0018183C3C18183C003C3C00003C3C7F
10430 DATA 00383C3E1C60FCFC00100C1C783868E800383C3E0C40FCFC
10440 DATA 001C3C78740E3F3F000828343C3C1E1F001C3C78600E3F3F
10450 DATA 0038380038387CFC0010103838103838003838000038 7CFC
10460 DATA 007070007070F8F800202070702020707000707000707 0F8F8
10470 DATA 1F1F1F0E00000000E0E0E0000000001F1F1F0E00000000
10480 DATA 3E3E3E3E1C0000001C1C1C1C000000003E3E3E3E1C000000
10490 DATA 7F7F7F7F7F3C00003B271F3F1C0000007F7F7F3F5C200000
10500 DATA FCFCFCFC38000000D8D8B8380000000DCDCBC7C38000000
10510 DATA 3F3F3F1F03000000F0F170300000003F2F371B00000000
10520 DATA FCFCFCFCF8000000B8D8D8D80000000FCFCFCFC38000000
10530 DATA F8F8F8700000000070707000000000F8F8F87000000000
10600 DATA 000E0E000E0E1F1F0004040E04040E0E000E0E000E0E1F1F
10610 DATA 001C3CFC70063F3F000830783C3C0E0F001C3CFC70061F2F
10620 DATA 0038380038387FFF0010103810103937003838003838 7FFF
```

```
10630 DATA 00383C7F0EF0F0F80010142E7C7CE0E000383C7F06E0E0E8
10640 DATA 001C3C7874430F3F00082834383C3F0F001C3C7874430F2F
10650 DATA 000107050E1FFEFE00010307070E3C9C000107010C1FFEFE
10660 DATA 0070F0807070F8F800202070202070700070B0007070F8F8
10670 DATA 1F1F1F0E00000000E0E0E000000000001F1F1F0E00000000
10680 DATA 3F3F3F3E1C000000171B1D1C00000000373B3D3E1C000000
10690 DATA FFFFFFFF7C300000AF9F7F7C38000000FFFF7F7C38000000
10700 DATA F8F8F8F870000000D0A0606000000000D8B878F870000000
10710 DATA 3F3F3F1F0000000017171B010000000037373B1D00000000
10720 DATA FCFCFCF800000000E8F0F8F800000000FCF4F8F800000000
10730 DATA F8F8F870000000000707070000000000F8F8F87000000000
10800 DATA 000E0E010E0E1F1F0004040E04040E0E000E0E010E0E1F1F
10810 DATA 000000E1E0077F7F000000C03373341F000000E1E0073F5F
10820 DATA 0000E0E0D8FCFFFF00004090B0181D800000E0E058BC5DE7
10830 DATA 0000387017F3FCFE00001020716F1FF80000387007F1F8FC
10840 DATA 001C3C78F4C30F3F00082834B83C3F0F001C3C78F4830F2F
10850 DATA 000000000007DFFF0000000000107F70000000000007DFF7
10860 DATA 00181E30BCDCF8F800080C1C70A8B03000181E30B4CCB878
10870 DATA 1F1F1F0F00000000E0E0D000000000001F1F1D0E00000000
10880 DATA FFFFF1E0C00000006FF3E04000000000EFFBF0E000000000
10890 DATA FFFFFFFF7F300000C79F3F7F3C000000CFBF3F7F3C000000
10900 DATA F8F8F0E0C0000000F0F0C00000000000F0F0C020C0000000
10910 DATA 3F3F3F1C00000000171B1D0000000000373B3D1C00000000
10920 DATA FFFFFFFE1C000000F8F8FC3800000000FAFDFE3C00000000
10930 DATA F8F0F0E0000000000070E0E000000000F8F0F0E000000000
11000 DATA 000000000E000F1E00000000000E070D000000000E000F1E
11010 DATA 00000000183853FF00000000101E8E4700000000183813CF
11020 DATA 00000000006073FF00000000004091010000000000607163
11030 DATA 0000000007F3FEE70000000000F163E0000000000781B6E7
11040 DATA 00000000B0B3230700000000200C1F0F00000000B0B30307
11050 DATA 0000003080E4FFFF00000040600007E1000000308044BFF5
11060 DATA 0000000E1C18FCF6000000081024B00E0000000E1408BC76
11070 DATA 1F1F1F0F00000000E0E0D000000000001F1F1D0200000000
11080 DATA FFFFFFE7C70000006192E04000000000759BE3E301000000
11090 DATA FFFFFFFCE00000009FFFFF7800000000FFFFFFF8C0000000
11100 DATA F8F0C0E000000000FCE0804000000000F0E080E000000000
11110 DATA 0F1E1C00000000000100000000000000D1E1800000000000
11120 DATA FFFF3F0F00000000FE7E1E0000000000FE7F1F0600000000
11130 DATA F8F0C080000007440000000000000000F860800000000000
11200 DATA 00000000000E010E0000F0500000000000000000000E010E
11210 DATA 00000000003853FF000000000108E46000000000003812CE
11220 DATA 00000000003FFFFF00000000000000FE00000000003F00FE
11230 DATA 000000000080E7E70000000000001C3C000000000080E7E7
11240 DATA 0000000000007067000000000000001F0000000000007067
11250 DATA 00000000000000FF0000000000000C0000000000000000C7
11260 DATA 0000000000053CF000000000002020000000000000053CF0
11270 DATA 1F1F1F0E00000000E0E00000000000001F1F110000000000
11280 DATA FFFFFCE0C00000006090E040000000007498E0E000000000
11290 DATA FFFFFE0000000000FFFF000000000000FFFFFE0000000000
11300 DATA CCCC000000000000F0F0000000000000CCCC000000000000
11310 DATA 0733330000000000180F0C00000000000733330000000000
11320 DATA FFFFFF3F000000003FFF1E0000000000BFFFFF0600000000
11330 DATA FBFBF0E000000000E4E4000000000000FBE0E00000000000
11400 DATA 000E0E000E0E1F1F0004040E04040E0E000E0E000E0E1F1F
11410 DATA 001C1C001C1C3E3F0008081C1C081C1C001C1C00001C3E3F
11420 DATA 00383C1E2E70FCFC0010142C3C3C78F800383C1E0670FCFC
11430 DATA 001C3C7C38063F3F000830381E1C1617001C3C7C30023F3F
11440 DATA 003C3C003C3C3CFE0018183C3C18183C003C3C00003C3CFE
11450 DATA 0038380038387C7C00101038381038380038380000387C7C
11460 DATA 007070007070F8F80020207020207070007070007070F8F8
11470 DATA 1F1F1F0E00000000E0E0E000000000001F1F1F0E00000000
11480 DATA 3F3F3F3F1F0000001D1B1B1B000000003F3F3F3F1C000000
11490 DATA FCFCFCF8C0000000F0F0E8C000000000FCF4ECD800000000
11500 DATA 3F3F3F3F1C0000001B1B1D1C000000003B3B3D3E1C000000
11510 DATA FEFEFEFEFE3C0000DCE4F8FC38000000FEFEFEFC3A040000
11520 DATA 7C7C7C7C38000000383838380000000007C7C7C7C38000000
11530 DATA F8F8F870000000000707070000000000F8F8F87000000000
11600 DATA 000E0F010E0E1F1F0004040E04040E0E000E0D000E0E1F1F
11610 DATA 0080E0A070F87F7F0080C0E0E0703C390080E08030F87F7F
11620 DATA 00383C1E2EC2F0FC0010142C1C3CFCF000383C1E2EC2F0F4
11630 DATA 001C3CFE700F0F1F000828743E3E0707001C3CFE60070717
11640 DATA 001C1C001C1CFEFF0008081C08089CEC001C1C001C1CFEFF
11650 DATA 00383C3F0E60FCFC00100C1E3C3C70F000383C3F0E60F8F4
11660 DATA 007070007070F8F80020207020207070007070007070F8F8
11670 DATA 1F1F1F0E00000000E0E0E000000000001F1F1F0E00000000
11680 DATA 3F3F3F3F1F000000170F1F1F000000003F2F1F1F00000000
11690 DATA FCFCFCF800000000E8E8D88000000000ECECDCB800000000
11700 DATA 1F1F1F1F0E000000B0506060000000001B1D1E1F0E000000
11710 DATA FFFFFFFF3E0C0000F5F9FE3E1C000000FFFFFE3E1C000000
11720 DATA FCFCFC7C38000000E8D8B83800000000ECDCBC7C38000000
```

```
11730 DATA F8F8F87000000000070707000000000000F8F8F87000000000
11800 DATA 0018780C3D3B1F1F001030380E150D0C0018780C2D331D1E
11810 DATA 0000000000E0FBFF000000000080E0EF0000000000E0FBEF
11820 DATA 00383C1E2FC3F0FC0010142C1D3CFCF000383C1E2FC1F0F4
11830 DATA 00001C0EE8CF3F7F000008048EF6F81F00001C0EE08F1F3F
11840 DATA 000007071B3FFFFF000002090D18B80100000707 1A3DBAE7
11850 DATA 0000008707E0FEFE00000003CCCE2CF80000008707E0FCFA
11860 DATA 00707080707 0F8F8002020702020707000707080707 0F8F8
11870 DATA 1F0F0F0700000000E07070000000000001F0F0F0700000000
11880 DATA FFFFFF7F38000001F1F3F1C000000005FBF7F3C00000000
11890 DATA FCFCFC3800000000E8D8B8000000000ECDCBC3800000000
11900 DATA 1F1F0F07030000000F0F0300000000000F0F03040300000
11910 DATA FFFFFFFFFE0C0000E3F9FCFE3C000000F3FDFCFE3C000000
11920 DATA FFFF8F0703000000F6CF070200000000F7DF0F0700000000
11930 DATA F8F8F8F00000000007070B000000000000F8F8B87000000000
12000 DATA 0000007038183F6F0000001008240D70000000 7028103D6E
12010 DATA 0000000C0127FFFF00000020600E08700000000C0122FDAF
12020 DATA 00000000DCDC4E0000000000430F8F00000000DCDC0E0
12030 DATA 0000000E0CF7FE70000000000F0687C0000000E0816DE7
12040 DATA 00000000006CEFF0000000000289800000000000068EC6
12050 DATA 0000000181CCAFF0000000087871E2000000000181CC8F3
12060 DATA 000000007000F078000000000070E0B000000007000F078
12070 DATA 1F0F030100000002E0200000000000001F06010000000000
12080 DATA FFFFFCF000000007F7E78000000000007FFEF86000000000
12090 DATA F078380000000000800000000000000000B0781800000000
12100 DATA 1F0F0307000000003F0701020000000000F0701070000000
12110 DATA FFFFFF3F07000000F9FFFF1E00000000FFFFFF1F03000000
12120 DATA FFFFFFE7E3000000864907020000000AED9C7C780000000
12130 DATA F8F8F8F00000000007070B000000000000F8F8B8400000000
12200 DATA 000000000A03C0F000000000040400000000000000A03C0F
12210 DATA 0000000000000FF000000000000003000000000000000E3
12220 DATA 0000000000000EE6000000000000000F80000000000000EE6
12230 DATA 000000000001E7E700000000000000383C000000000001E7E7
12240 DATA 000000000FCFFFF00000000000000007F000000000FC007F
12250 DATA 000000000001CCAFF0000000000087162000000000001C4873
12260 DATA 00000000000708070000000000000F0A0000000000000708070
12270 DATA DFDF0F070000000027270000000000000DF0707000000000
12280 DATA FFFFFFFC00000000FCFF7800000000000FDFFFF6000000000
12290 DATA E0CCCC0000000000018F030000000000E0CCCC000000000
12300 DATA 3333000000000000000F0F000000000000333300000000000
12310 DATA FFFF7F0000000000FFFF000000000000FFFF7F0000000000
12320 DATA FFFF3F0703000000609070200000000002E1907070000000
12330 DATA F8F8F870000000000707000000000000000F8F88800000000
12400 DATA 000E0E000E0E1F1E0004040E04040B0D000E0E000E0E1F1E
12410 DATA 000E0E000E0F1F1E0004040E04040D0D000E0E000E0F1D1E
12420 DATA 00707000B8D8F8F80020207070A0903000707000B0C8B878
12430 DATA 00707018B8D8F8F80020207070A0B03000707018B0C8B878
12440 DATA 0070700070 70F8F80020207020207000007070007070F8F8
12450 DATA 003838003F33FCFE00101038390F1FF80038380007 31F8F8
12460 DATA 0038380038387EE70010103838 10361E0038380000387EE7
12470 DATA 1F1F1F0F00000000E0E0D00000000001F1F1D0E00000000
12480 DATA 1FCFCF0F00000000E303F00000000001FCFCF000000000
12490 DATA F8F8F87000000000070707000000000000F8F8F87000000000
12500 DATA F8F8F8700000000070F0700000000000F8F8F87000000000
12510 DATA FBFBF8F00000000E4E4000000000000FBE0E0000000000
12520 DATA FCFCFCFC38000000F0E0983800000000F4E49C7C38000000
12530 DATA F8FCFC7C38000000FCC83838000000000F0CC3C7C38000000
12600 DATA 0038380038387C7C0010103838103838003838 0000387C7C
12610 DATA 000E0E080D1B1F1F00040406 0E050D0C000E0E080D131D1E
12620 DATA 000E0E000E0E3F6F0004040E04040D70000E0E000E0E3D6E
12630 DATA 000E0E000E0E1F1F0004040E04040E00000E0E000E0E1F1F
12640 DATA 0070700070F0F8780020207020 20D0B00070700070F0F878
12650 DATA 0070700070F0F87B0020207020 20D0B40070700070F0F87B
12660 DATA 00383C0F06E4FCFC0010143E7C7CE8E800383C0F06E0ECEC
12670 DATA CCCC7C7C38000000F0F0383800000000CCCC7C7C38000000
12680 DATA 1F1F1F0E00000000E0E0E00000000001F1F1F0E00000000
12690 DATA 1F1F1F0F000000002E020C00000000001F161D0E00000000
12700 DATA DFDF1F0F000000002727000000000000DF071708000000000
12710 DATA F8F8F8F00000000007070B000000000000F8F8B87000000000
12720 DATA FBF8F870000000000747070000000000000F8F8F87000000000
12730 DATA FCFCFC7C38000000D8B838380000000000DCBC7C7C38000000
12800 DATA 001C3CF0742E3F3F0008287C383C1717001C3CF064023737
12810 DATA 001C1C00FCCC3E7F0008081C9CF0F81F001C1C00E08C1E3F
12820 DATA 001C1C001C1C7EE70008081C1C085C78001C1C00001C7EE7
12830 DATA 001C1C001C1C3E3E0008081C1C081C1C001C1C00001C3E3E
12840 DATA 000000000000000000000001E6DDBD96C0000001E6DDBD96C
12850 DATA 00181818000018001 83C6624183CA5DB183C6624183CA5DB
12860 DATA 00000000000000000000000078B6DB9B3600000078B6DB9B36
12870 DATA 3F3F3F3E1C0000001B1D1C1C00000003B3D3E3E1C000000
12880 DATA 3F3F3F3F1C0000000F0F131C000000002F2F333C1C000000
```

```
12890 DATA 1F3F3F3E1C0000003F131D1C000000000F333D3E1C000000
12900 DATA 73733E3E1C0000000F0F1C1C0000000073733E3E1C000000
12910 DATA 00000000000800006633190603151F006633190603151F00
12920 DATA 18000000008100007E18A55A245AFF007E18A55A245AFF00
12930 DATA 000000000010000066CC9860C0A8F80066CC9860C0A8F800
49999 '
50000 ' ------------------------------------------
50010 '  PCG EDITOR   by   ELDER MAN  Vr.1.2
50020 ' ------------------------------------------
50030 TEMPO 300:DEFINT A-Z
50040 WIDTH80:COLOR7,0:SCREEN0,0,0
50050 CGEN0:WINDOW:CLEAR:BEEP:LAYER 1,2,3,4:CANVAS 5,1,0
50060 DIM A%(55,15),B%(2),ST$(32)
50070 FOR I=0 TO 56:C=1
50074 SCREEN 0,0,2
50075 IF (I MOD 8)=0 THEN C=5:SCREEN 0,0,1
50080 LINE (I*8,0)-(I*8,16*8),PSET,C:NEXT
50082 FOR I=0 TO 16:C=1
50084 SCREEN 0,0,2
50086 IF (I MOD 8)=0 THEN C=5:SCREEN 0,0,1
50090 LINE (0,I*8)-(56*8,I*8),PSET,C:NEXT
50110 LOCATE 60,0:PRINT"* PCG EDITOR *"
50120 LOCATE 60,6:PRINT"8 = UP"
50130 LOCATE 60,7:PRINT"4 = LEFT"
50140 LOCATE 60,8:PRINT"5 = SET"
50150 LOCATE 60,9:PRINT"6 = RIGHT"
50160 LOCATE 60,10:PRINT"2 = DOWN"
50170 LOCATE 60,11:PRINT"C = CLS"
50180 LOCATE 60,12:PRINT"D = DELETE"
50190 LOCATE 60,13:PRINT"G = GENERATE"
50200 LOCATE 60,14:PRINT"R = RECALL"
50210 LOCATE 60,16:PRINT"COLOR (0-7)"
50220 LOCATE 60,18:PRINT"CHR$ NO.(1-18)"
50250 LOCATE 0,0:X=0:Y=0
50260 S$=INKEY$(0)
50270 IF S$="8" THEN Y=Y-1
50280 IF S$="4" THEN X=X-1
50290 IF S$="5" GOSUB 51000
50300 IF S$="6" THEN X=X+1
50310 IF S$="2" THEN Y=Y+1
50320 IF S$="C" THEN 50000
50330 IF S$="D" THEN 53000
50340 IF S$="G" THEN 54000
50350 IF S$="R" THEN 55000
50360 IF X<0 THEN X=0
50370 IF X>55 THEN X=55
50380 IF Y<0 THEN Y=0
50390 IF Y>15 THEN Y=15
50400 GET@ (X,Y)-(X,Y),B%
50405 LOCATE X,Y:PRINT " ":MUSIC"R0"
50410 LOCATE X,Y:PRINT "■":MUSIC"R0"
50420 PUT@ (X,Y)-(X,Y),B%
50450 COLOR7:GOTO 50260
51000 LOCATE X,Y:PRINT"■";:MUSIC"V13O5B0"
51001 S$=INKEY$:IF S$<>"" THEN 51001
51002 LOCATE60,17:PRINT "INPUT=";:S$=INKEY$(1):C=VAL(S$):MUSIC"V13O6C0"
51005 IF C<0 OR C>7 THEN 51000
51010 COLOR C:LOCATE X,Y:PRINT"■"
51020 A%(X,Y)=C:RETURN
53000 S$=INKEY$:IF S$<>"" THEN 53000
53005 BEEP:LOCATE60,20:PRINT "REARY? Y/N=";:S$=INKEY$(1):BEEP
53010 IF S$="Y" THEN DELETE 50000-
53020 IF S$="N" THEN 53040
53030 GOTO 53000
53040 LOCATE25,20:PRINT"
53050 GOTO 50260
54000 S$=INKEY$:IF S$<>"" THEN 54000
54010 BEEP:LOCATE60,19:INPUT "INPUT=",CH:MUSIC"V13O5G0"
54015 IF CH<1 OR CH>18 THEN 54010
54017 L=9800+CH*200
54020 WIDTH 80:O=0
54030 FOR P=0 TO 8  STEP 8
54040 FOR Q=0 TO 48 STEP 8
54045 PRINT L+O;"DATA ";:V=0
54050 FOR T=1 TO 3
54060 FOR S=P TO P+7:U=0:V=V+1
54070 FOR R=Q TO Q+7:U=U+1
54080 ON U GOSUB 54400,54410,54420,54430,54440,54450,54460,54470
54085 ON T GOSUB 54500,54600,54700
54090 NEXT
```

```
54095 IF D<16 THEN D$="0"+HEX$(D) ELSE D$=HEX$(D)
54097 IF V=24 THEN PRINT D$:D=0:GOTO 54100
54098 PRINT D$;:D=0
54100 NEXT:NEXT:O=O+10:NEXT:NEXT
54110 END
54400 E=128:RETURN
54410 E=64:RETURN
54420 E=32:RETURN
54430 E=16:RETURN
54440 E=8:RETURN
54450 E=4:RETURN
54460 E=2:RETURN
54470 E=1:RETURN
54500 IF A%(R,S)=1 OR A%(R,S)=3 OR A%(R,S)=5 OR A%(R,S)=7 THEN D=D+E
54510 RETURN
54600 IF A%(R,S)=2 OR A%(R,S)=3 OR A%(R,S)=6 OR A%(R,S)=7 THEN D=D+E
54610 RETURN
54700 IF A%(R,S)>3 THEND=D+E
54710 RETURN
55000 S$=INKEY$:IF S$<>"" THEN 55000
55010 BEEP:LOCATE60,19:INPUT "INPUT=",CH:MUSIC"V1305A0"
55015 IF CH<1 OR CH>28 THEN 55010
55020 ON CH RESTORE 10000,10200,10400,10600,10800,11000,11200,11400,11600,11800,
12000,12200,12400,12600,12800,13000,13200,13400
55030 FOR I=30 TO 43:READ A$
55040 DEF CHR$(I)=HEXCHR$(A$):NEXT
55050 CGEN1:LOCATE 65,2:PRINT#0 CHR$(30)+CHR$(31)+CHR$(32)+CHR$(33)+CHR$(34)+CHR
$(35)+CHR$(36)
55060 LOCATE 65,3:PRINT#0 CHR$(37)+CHR$(38)+CHR$(39)+CHR$(40)+CHR$(41)+CHR$(42)+
CHR$(43)
55080 CGEN0:LOCATE 60,21:PRINT "CORRECT ? Y/N":BEEP
55090 S$=INKEY$:IF S$="" THEN 55090
55095 IF S$="Y" THEN BEEP:GOTO 58000
55100 GOTO 50010
58000 W=0:L=0
58010 FOR I=30 TO 43:CG$=CGPAT$(I):C=1
58020 FOR J=0 TO 23
58030 NA=ASC(MID$(CG$,J+9,1))
58040 IF NA>=128 THEN A%(W,K+L)=A%(W,K+L)+C:NA=NA-128
58050 IF NA>=64 THEN A%(W+1,K+L)=A%(W+1,K+L)+C:NA=NA-64
58060 IF NA>=32 THEN A%(W+2,K+L)=A%(W+2,K+L)+C:NA=NA-32
58070 IF NA>=16 THEN A%(W+3,K+L)=A%(W+3,K+L)+C:NA=NA-16
58080 IF NA>=8 THEN A%(W+4,K+L)=A%(W+4,K+L)+C:NA=NA-8
58090 IF NA>=4 THEN A%(W+5,K+L)=A%(W+5,K+L)+C:NA=NA-4
58100 IF NA>=2 THEN A%(W+6,K+L)=A%(W+6,K+L)+C:NA=NA-2
58110 IF NA>=1 THEN A%(W+7,K+L)=A%(W+7,K+L)+C
58120 K=K+1:IF K>7 THEN K=0
58125 IF J>=7 THEN C=2
58130 IF J>=15 THEN C=4
58135 NEXT:W=W+8:IF W>55 THEN W=0
58140 IF I>35 THEN L=8
58160 NEXT
58200 FOR I=0 TO 15:FOR J=0 TO 55
58210 LOCATE J,I:COLOR A%(J,I):PRINT "■"
58220 NEXT:NEXT:GOTO 50250
```

●画面のレイアウト

ボーリングの臨場感を出すために，実際にボールをリリースしたプレイヤーの位置からみえる角度で，レーンやピンのレイアウトを決めます。ここで，ボーリングのゲームに登場するいろいろなものの，プログラム上の割りあてを決定しなければなりません。ここまでの段階では，ストライクなどのピンアクションの絵がPCGで作られているだけですから，さらに投球されるボールや，レーン，スコア表などの作り方を決めておく必要があります。レーンとボールはグラフィックで表現するため，動くボールによって背景のレーンが消えてしまわないようにしなければなりません。また，ピンはレーンより優先で前面に描き，ボールはピンよりも前面にする必要があります。

これらのことから，プログラムでは，CANVAS命令，LAYER命令を使用し，単色3画面のマルチ画面モードとします。画面の優先順位と役割りとを表1にまとめました。

プログラムでは，CANVAS 2, 6, 1：

表1 テキスト画面とマルチ画面の役割

画　面	優先順位	色	役　　割〔使用するプログラム命令〕
テキスト	2	白	ゲームのタイトル，スコアの表示，ピンの表示（PCG）〔PRINT 文〕
グラフィック1	4	赤	スコアを表示する表の線〔LINE 文〕
グラフィック2	3	黄	レーンを描く〔LINE 文〕
グラフィック3	1	青	ボールを描く，残りピンをクリアーする〔PATTERN 文または PUT@文〕

LAYER 2，4，3，1と書きます。（リスト2の8460行）

知っている人も多いでしょうが，ボーリングのレーンは30枚の板でできていて，5枚目ごとにスパットと呼ばれる三角の目印がつけられています。この辺のところの芸も細かく演出します。

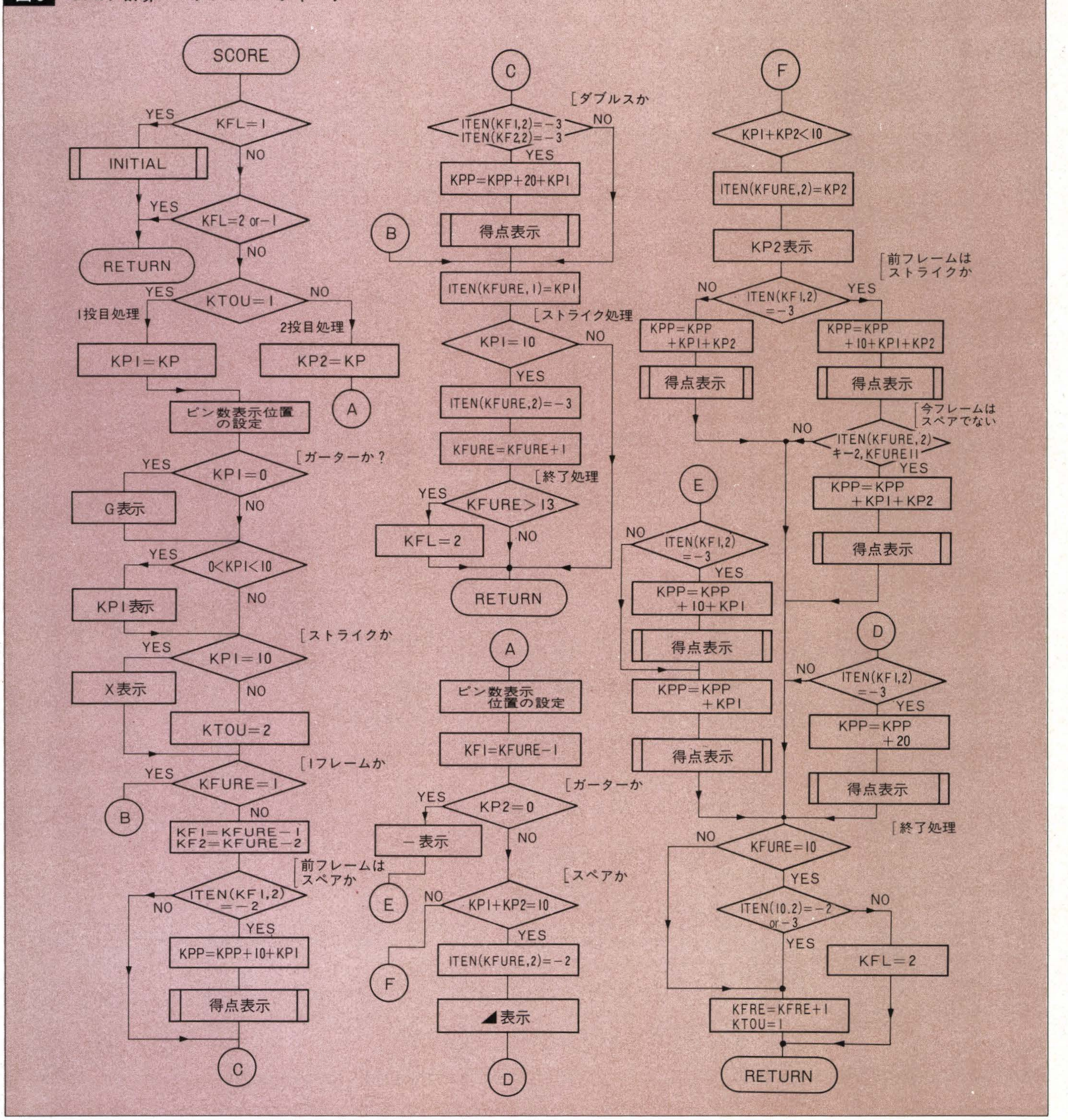
画面の一番上にはゲームのタイトルを表示します。作者の名前もお忘れなく。

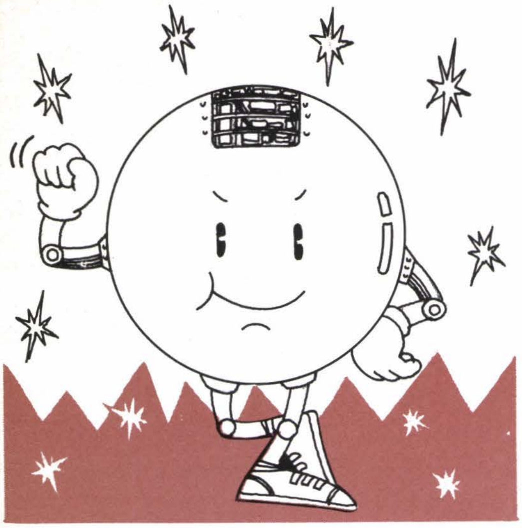
その下に，プレイヤー2人分のスコア表を描きます。今回と次回では1人分でいいのですが，最終バージョンでは2人分のものを使用します。ゲームは競う相手があれば，一層おもしろくなります。マイコンゲームであれば，それをコンピュータにやらせない手はありませんネ。画面の中にみられる四角いボックスの用途については次回のお楽しみということにします。

●スコア計算ルーチンの作成

今回のメインプログラム（リスト2）は，画面のレイアウトと得点計算ルーチンだけです。適当なピン数を入力して，ちゃんと計算があっているか確認してみてください。プログラムの8290行から8590行までが得点計算サブルーチンになっているので，独立して他のプログラムに転用可能です。

図3　スコア計算ルーチンフローチャート





スコア計算ルーチンのフローチャートを図3に示しました。かなり複雑なフローチャートになってしまいましたが，実際にボーリングの点数のつけ方というのは人間的で，コンピュータ向きではないようです。とくにストライクの得点計算や第10フレームの変則的なスコア表など，どうしても IF 文が多いプログラムになってしまいました。

プログラムの変数について，コメントしておきます。

ITEN (12, 2)：各フレームのストライクかスペアかの記録用

KFL：エラー処理フラグで，1のとき画面表示，2のときゲーム終了，−1でエラー発生（10本以上倒してしまう）

KFURE：投球フレーム数

KTOU　：第1投か第2投か

KP　　：入力する得点（順々に入力）

KP1，KP2：第1投，第2投の得点

KPP　　：表示する合計得点

リスト 2

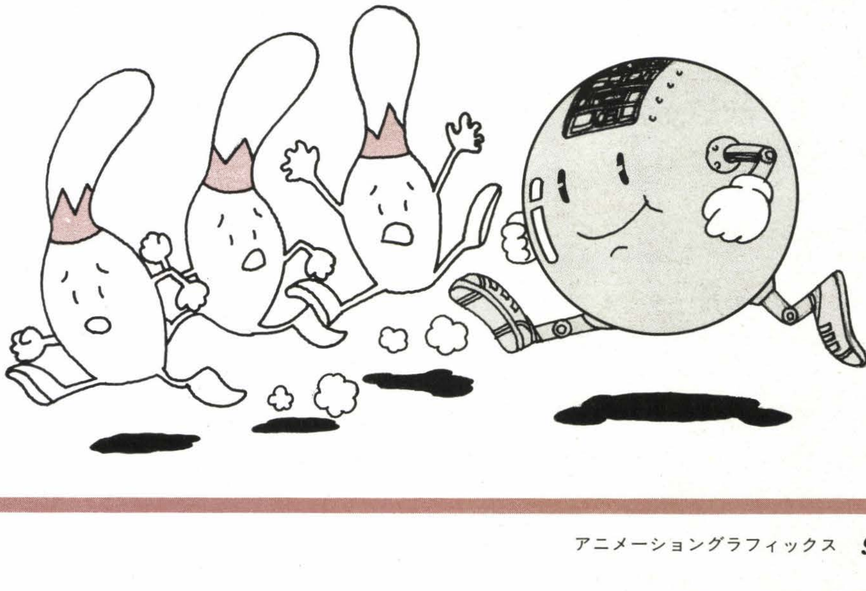
```
10 '=============================================================
20 '
30 ' BOWLING (MAIN PROGRAM)  Copyright  1983  by  ELDER MAN  Vr.1.0
40 '
50 '=============================================================
60 GOTO 8000
6000 '
6010 ' ====== POCKET HIT ======
6020 '
6040 CGEN1:LOCATE36,14:PRINT"?@ABCDE":LOCATE36,15:PRINT"FGHIJKL"
6050 PAUSE1:LOCATE36,14:PRINT"MNOPQRS":LOCATE36,15:PRINT"TUVWXYZ"
6060 PAUSE1:LOCATE36,14:PRINT"[¥]^_`a":LOCATE36,15:PRINT"bcdefgh"
6070 PAUSE1:LOCATE36,14:PRINT"ijklmno":LOCATE36,15:PRINT"pqrstuv"
6080 PAUSE1:LOCATE36,14:PRINT"wxyz{|}":LOCATE36,15:PRINT"¯π___▄▄"
6090 CGEN:PAUSE5:RETURN
6100 '
6110 ' ====== BROOKLYN ======
6120 '
6140 CGEN1:LOCATE36,14:PRINT"▄▄█ ▌▌▌":LOCATE36,15:PRINT"███/─|┴"
6150 PAUSE1:LOCATE36,14:PRINT"┬┤├┼┘└":LOCATE36,15:PRINT"┌╮╰╯╱╲ "
6160 PAUSE1:LOCATE36,14:PRINT"｡｢｣､･ｦｧ":LOCATE36,15:PRINT"ｨｩｪｫｬｭｮ"
6170 PAUSE1:LOCATE36,14:PRINT"ｯｰｱｲｳｴｵ":LOCATE36,15:PRINT"ｶｷｸｹｺｻｼ"
6180 PAUSE1:LOCATE36,14:PRINT"ｽｾｿﾀﾁﾂﾃ":LOCATE36,15:PRINT"ﾄﾅﾆﾇﾈﾉﾊ"
6190 CGEN:PAUSE5:RETURN
8000 '
8010 ' ======= SCORE CAL ========
8020 '
8030 WIDTH 80:DIM ITEN(12,2)
8040 KFL=1:GOSUB 8120:GOSUB 8600
8045 CGEN1:LOCATE36,14:PRINT"#$%&'()":LOCATE36,15:PRINT"*+,-./0":CGEN
8048 IF KTOU=1 THEN LOCATE36,12:PRINT"1        " ELSE LOCATE36,12:PRINT"        2"
8050 LOCATE9,20:PRINT"  ":LOCATE1,20:INPUT "SCORE = ",KP
8055 IF KTOU=2 AND KP1+KP>10 THEN BEEP:GOTO8050
8060 IF KP=10 THEN IF RND<.5 GOSUB6000 ELSE GOSUB 6100
8070 GOSUB8120:GOTO 8045
8120 IF KFL=1 THEN 8440 ELSE IF KFL=2 OR KFL=-1 THEN RETURN
8130 IF KTOU=1 THEN KP1=KP ELSE KP2=KP:GOTO 8290
8140 '
8150 ' ====== CAL OF 1 TOU ======
8160 '
8170 IF KFURE>10 AND ITEN(10,2)=-3 THEN LOCATE 40+2*KFURE,3  ELSE LOCATE 20+4*KF
URE,3
8180 IF KP1=0 THEN PRINT "G";
8190 IF KP1>0 AND KP1<10 THEN PRINT USING "#";KP1;
8200 IF KP1=10 THEN IF KFURE<10 THEN PRINT"  X"; ELSE PRINT"X"; ELSE KTOU=2
8210 IF KFURE=1 THEN 8250
8220 KF1=KFURE-1:KF2=KFURE-2
8230 IF ITEN(KF1,2)=-2 THEN KPP=KPP+10+KP1:GOSUB 8550
8240 IF ITEN(KF1,2)=-3 AND KF2>0 THEN IF ITEN(KF2,2)=-3 THEN KPP=KPP+20+KP1:KF1=
KF2:GOSUB 8550
8250 ITEN(KFURE,1)=KP1
8260 IF KP1=10 THEN ITEN(KFURE,2)=-3:KFURE=KFURE+1:IF KFURE>13 THEN KFL=2
8270 RETURN
8280 '
8290 ' ======= CAL OF 2 TOU ======
8300 '
8310 IF KFURE>10 AND ITEN(10,2)=-3 THEN LOCATE 42+2*KFURE, 3 ELSE LOCATE 22+4*KF
```

```
URE,_3
8320 KF1=KFURE-1
8330 IF KP2=0 THEN PRINT"-";:GOTO 8390
8340 IF KP1+KP2=10 THEN ITEN(KFURE,2)=-2:PRINT"◢";:GOTO8380
8350 IF KP1+KP2<10 THEN ITEN(KFURE,2)=KP2:PRINT USING"#";KP2;ELSE KFL=-1:RETURN
8360 IF ITEN(KF1,2)<>-3 THEN KPP=KPP+KP1+KP2:KF1=KFURE:GOSUB 8550:GOTO 8410
8370 KPP=KPP+10+KP1+KP2:GOSUB 8550:IF ITEN(KFURE,2)<>-2 AND KFURE<>11 THEN KPP=K
PP+KP1+KP2:KF1=KFURE:GOSUB 8550:GOTO 8410:ELSE 8410
8380 IF KF1>0 AND ITEN(KF1,2)=-3 THEN KPP=KPP+20:GOSUB 8550:GOTO 8410 ELSE 8410
8390 IF KF1>0 AND ITEN(KF1,2)=-3 THEN KPP=KPP+10+KP1:GOSUB 8550
8400 KPP=KPP+KP1:KF1=KFURE:IF KFURE<11 GOSUB 8550
8410 IF KFURE=10 THEN IF ITEN(10,2)=-2 OR ITEN(10,2)=-3 THEN KFURE=11:KTOU=1:RET
URN ELSE KFL=2:RETURN
8420 KFURE=KFURE+1:KTOU=1:RETURN
8430 '
8440 ' ====== LINE & INITIALIZE ======
8450 LOCATE4,0:PRINT"****      BOWLING  GAME      1983.8.28       Vr.1.0   by  ELDE
R MAN      ****"
8460 CANVAS2,6,1:LAYER2,4,3,1:IY=15:IX=82:LOCATE14,2:PRINT"NAME";
8470 FOR I=1 TO 9:LOCATE 20+4*I,2 :PRINT I;:NEXT:PRINT"  10   TOTAL"
8480 SCREEN0,0,1:LINE(IX,IY)-(IX+484,IY+41),PSET,B
8485 LINE(0,IY)-(IX-4,IY+41),PSET,B:LINE(IX+488,IY)-(639,IY+41),PSET,B
8490 LINE(IX,IY+8)-(IX+484,IY+8),PSET:LINE(IX+104,IY+16)-(IX+440,IY+16),PSET
8495 LINE(IX,IY+24)-(IX+484,IY+24),PSET:LINE(IX+104,IY+32)-(IX+440,IY+32),PSET
8500 LINE(IX+440,IY)-(IX+440,IY+41),PSET
8510 FOR I=1 TO 10:LINE(IX+72+32*I,IY)-(IX+72+32*I,IY+41),PSET:NEXT
8520 FOR I=1 TO 20 LINE(IX+104+16*I,IY+8)-(IX+104+16*I,IY+16),PSET:NEXT
8525 FOR I=1 TO 20:LINE(IX+104+16*I,IY+24)-(IX+104+16*I,IY+32),PSET:NEXT
8530 KTOU=1:KFURE=1:KPP=0:KFL=0:ITEN(10,2)=0:RETURN
8540 '
8550 ' ====== SCORE DISPLAY ======
8560 '
8570 LOCATE20+4*KF1,4
8580 IF KF1=10 THEN PRINT USING "##### ####";KPP;KPP;:KFL=2:RETURN
8590 PRINT USING "###";KPP;:RETURN
8600 '
8610 ' ====== LANE DRAWING ======
8620 '
8630 SCREEN,,2:LINE(0,64)-(639,64),PSET
8640 LINE(277,70)-(356,70)-(322,104)-(310,104)-(277,70)
8670 LINE(277,70)-(356,104),PSET,B
8680 LINE(270,64)-(270,120)-(70,199):LINE(570,199)-(363,120)-(363,64)
8690 LINE(288,120)-(116,199),PSET:LINE(518,199)-(341,120),PSET
8700 FORI=1 TO 29:LINE(117+13.2*I,199)-(288+1.8*I,120),PSET:NEXT
8710 LINE(225,170)-(219,172)-(224,172)-(225,170)
8720 LINE(271,168)-(268,170)-(271,170)-(271,168)
8730 LINE(315,166)-(313,168)-(317,168)-(315,166)
8740 LINE(359,168)-(359,170)-(363,170)-(359,168)
8750 LINE(405,170)-(411,172)-(406,172)-(405,170)
8800 LINE(0,70)-(200,144),PSET,B
8810 LINE(0,151)-(100,183),PSET,B
8820 LINE(400,70)-(504,120),PSET,B
8830 LINE(518,70)-(639,175),PSET,B
8850 RETURN
```

●次回予告

いよいよ次回は，１人でプレイできるようにします。投球方法やボールのころがりなどのプログラムテクニックを楽しみにしていてください。

このシリーズでは，プログラム入門というよりも，ゲームプログラムをどのように作るのかというところにポイントを置いています。御意見をいただきたいと思ってますので，編集室あてにお便りをお送りください（アンケートはがきでも結構だと思います)。



X1強化実践教室2

インターフェースの製作

●ボードの組み立て方

井内 秀則

はじめに

先月号で予告しましたように，今月から「X 1」強化実践教室第一弾として,「シリアル・パラレル・カウンタインターフェース」を製作していきます。本講座では，初心者にもハードウェアを段階的に理解してもらえるように，主たる用語，難解な用語には脚注という形で説明していきますから，そちらのほうも参照してください。

マイコンを外部の周辺機器とつなぐ場合に必要なインターフェースをワンボードに実装したものを製作しますので，このボードを組み上げたときには，マイコンをコントローラとして使うのにどのような回路を付ければよいかが，ご理解いただけると思います。ハードウェアの製作は，何といっても経験が必要です。何事もそうですが，一度自分の手で作ってみれば自信がつくものです。二度目に同様の回路を作るときには一度目よりも早く作ることができるでしょう。

それでは，具体的に回路図を示しながら詳細について説明していきましょう。

まず，RS-232Cによるシリアル通信に必要なボー・レート・クロックの作成方法から説明していきます。

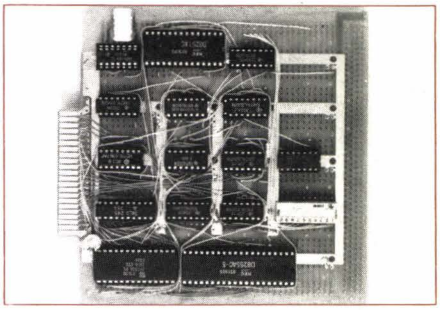
1.ボー・レート・クロックの作製

ボー・レート・クロックの作製方法について述べる前に，このクロックを入力するシリアルインターフェースICは通信の基準となるクロックの16倍，または64倍のクロックを必要とすることに注意してください。

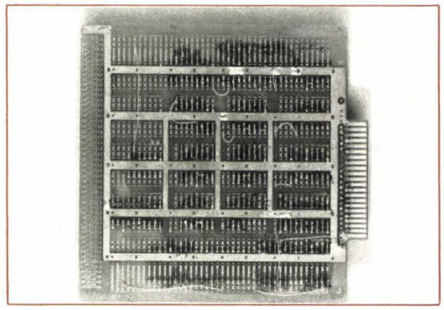
たとえば，9600ボーの通信速度の場合には，9600×16=153,600つまり153.6KHzのクロックをSIOのクロック入力端子 $\overline{TXC}$, $\overline{RXC}$ に供給する必要があります。

このためのボー・レート・クロックの作製方法には，専用のICを用いる方法，74LS163などの同期プリセット・同期クリアのいずれかの端子を持つカウンタICを用いる方法，さらにはSY6551などのシリアルインターフェスICのように，ソフトウェアでボーレートを選択できるようなICを用いる方法などが考えられます。

今回は，入手の容易さや価格の安さなどを考慮して，8251AをSIOとして使うことにしましたので，カウンタICでボー・レート・クロックを作製することになります。

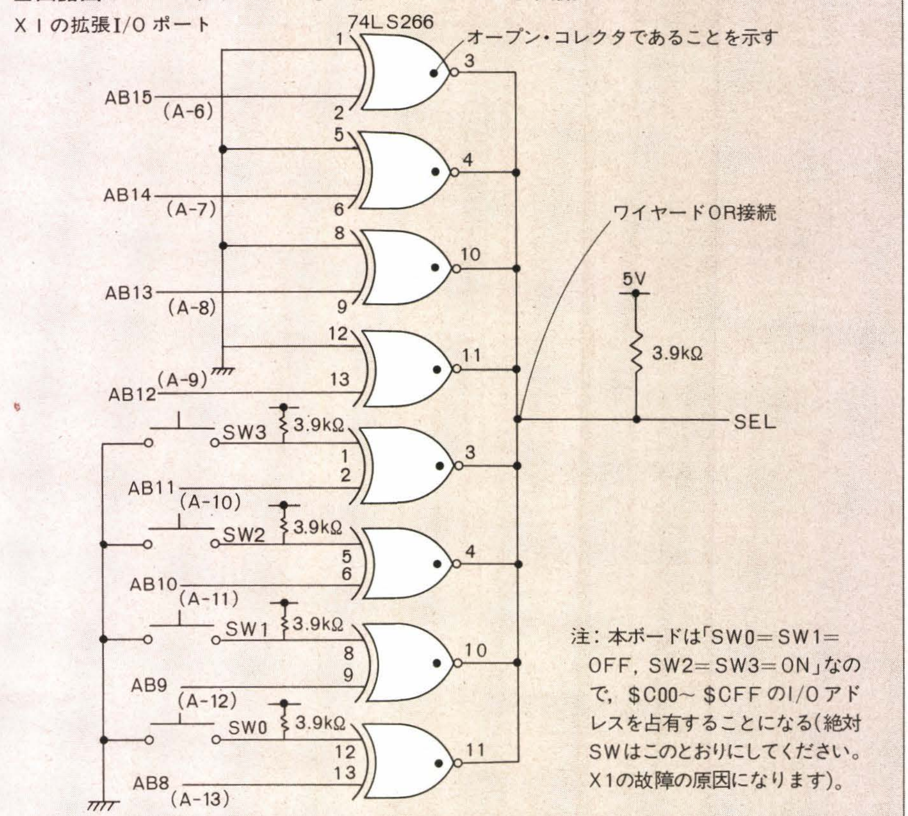


完成したシリアル・パラレル・カウンター インターフェース

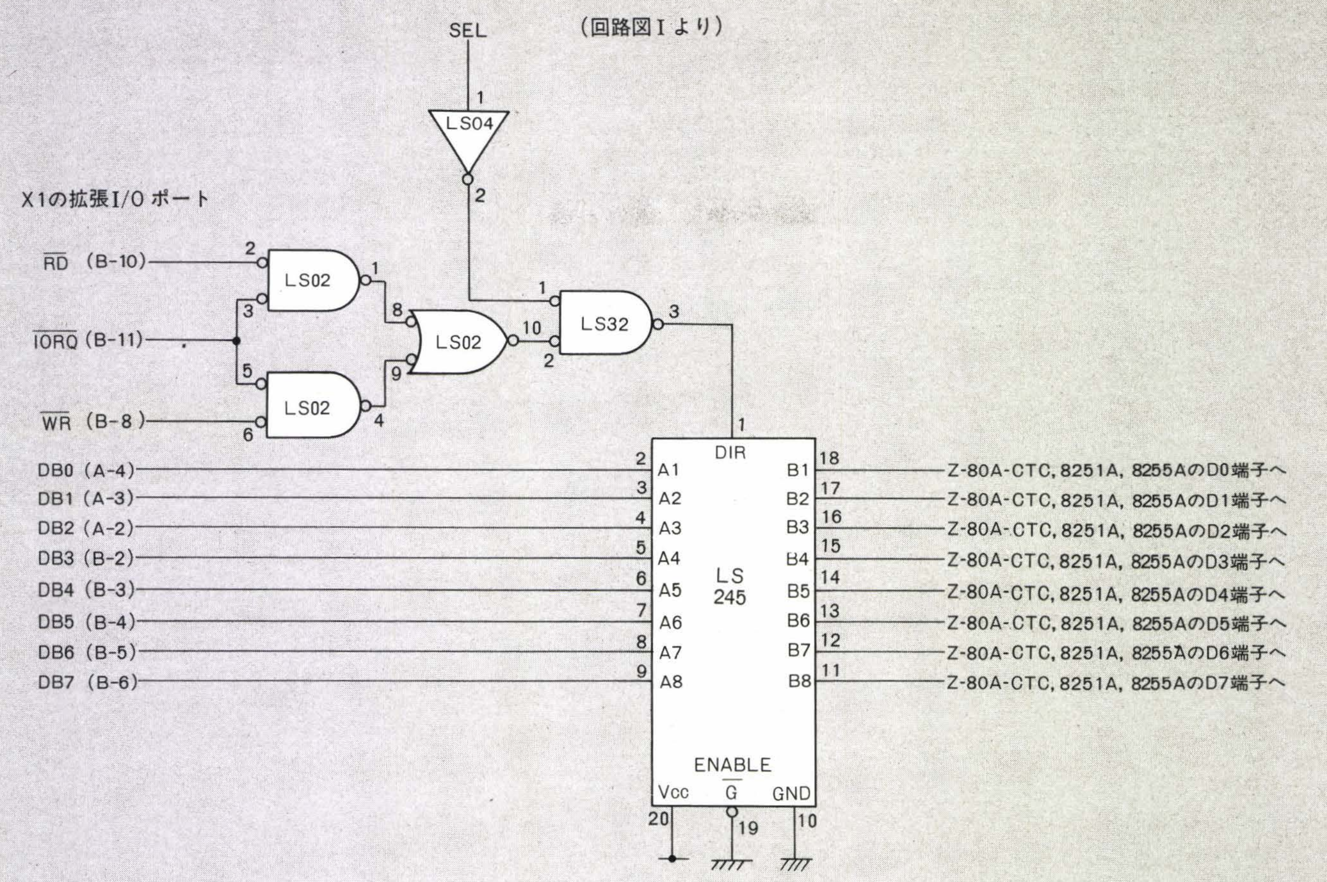


注)．コネクタ（DB25S）はまだついていません。

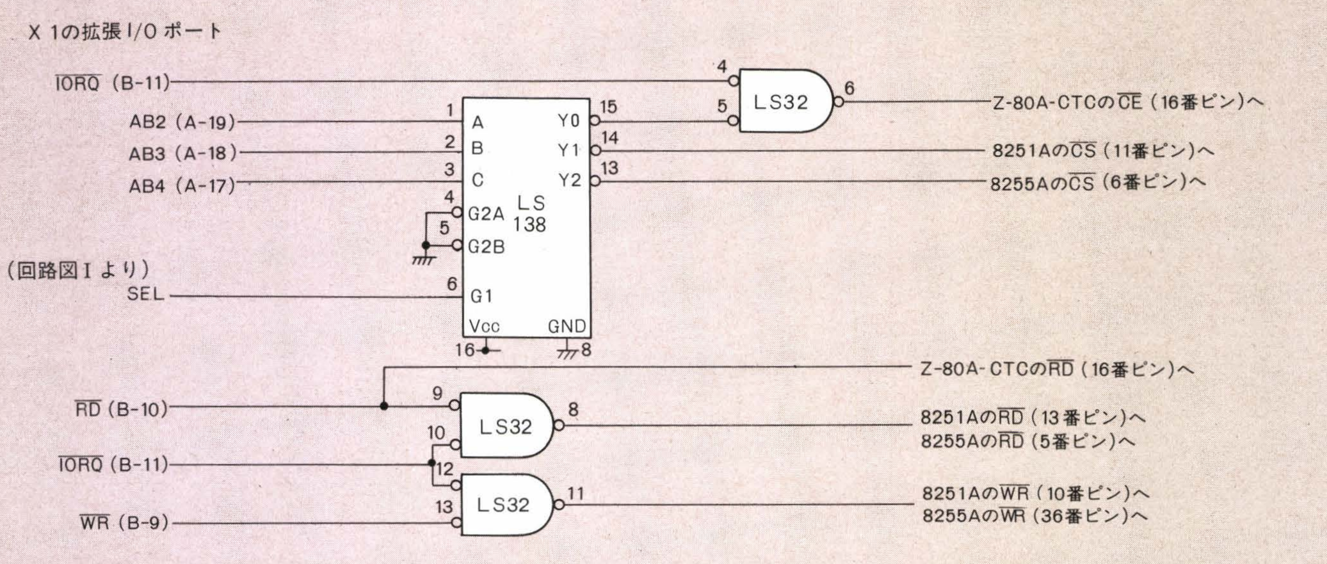
全回路図 I　プログラマブルなアドレス・デコード回路



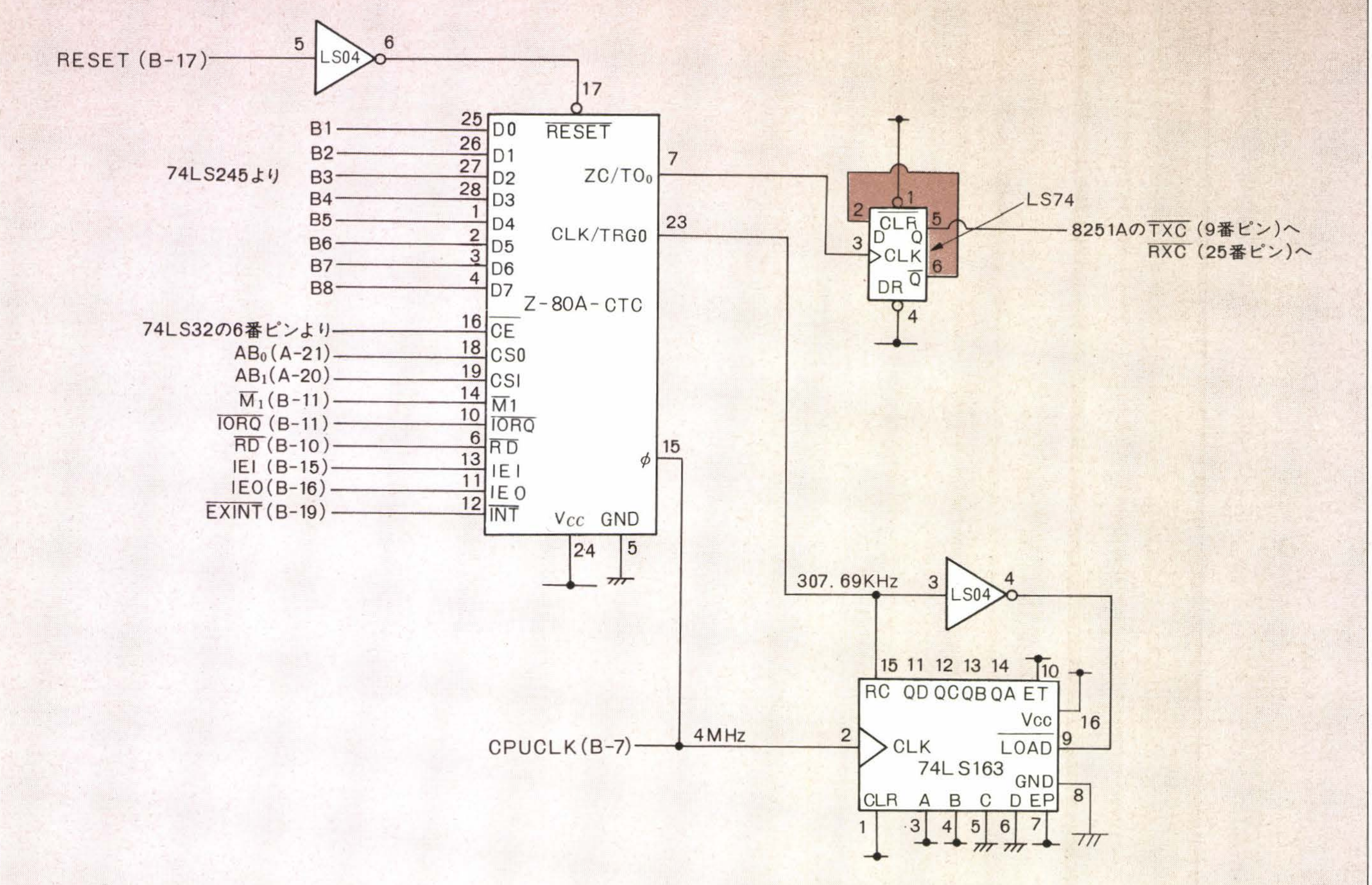
全回路図II　データ・バスのバッファリング回路



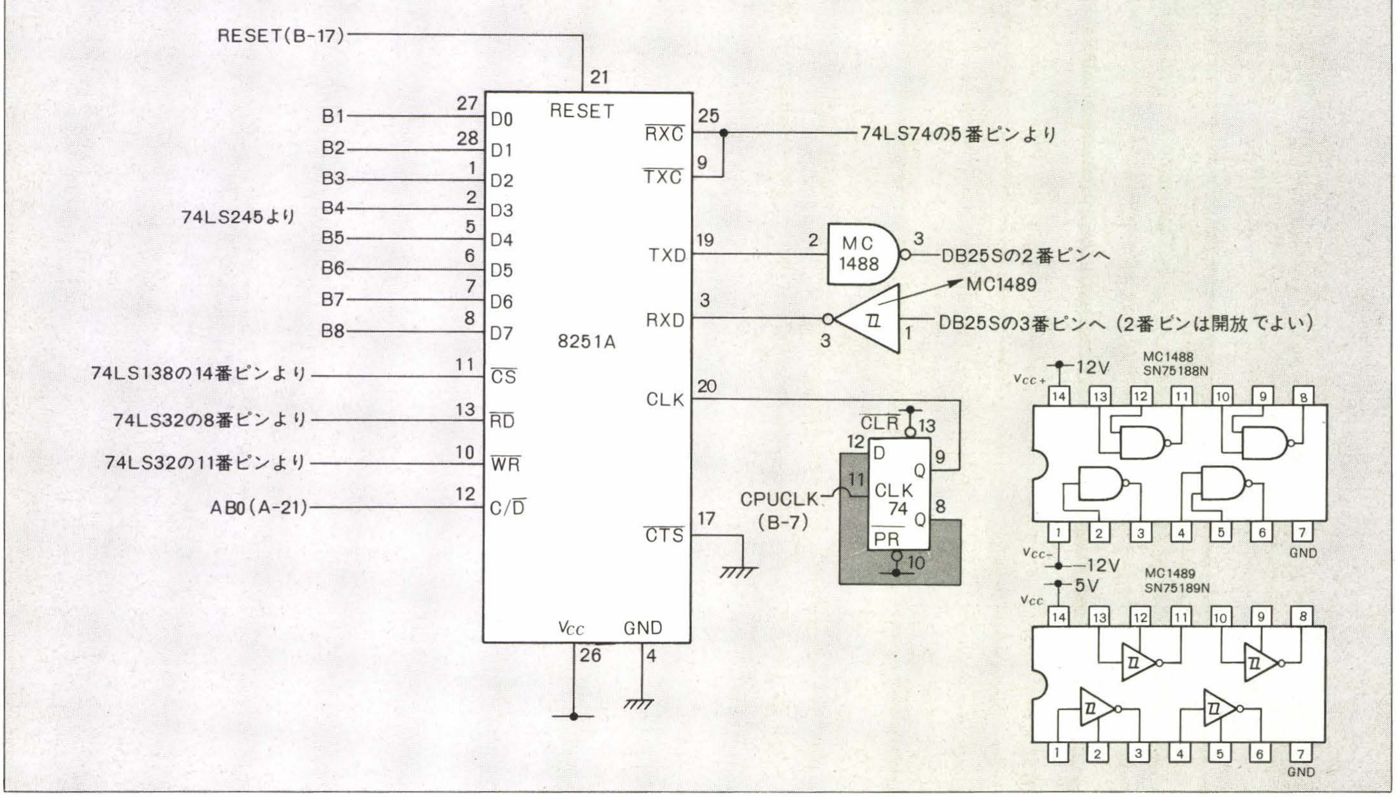
全回路図III　周辺LSIのチップセレクト回路とREAD・WRITEコントロール回路



全回路図IV　Z-80A・CTC周辺の回路



全回路図V　8251A周辺の回路



全回路図VI　8255Aの周辺回路

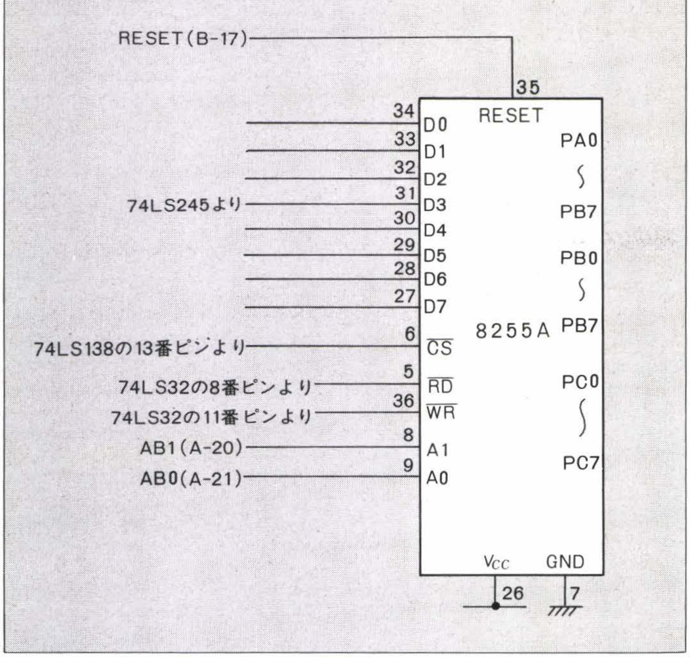
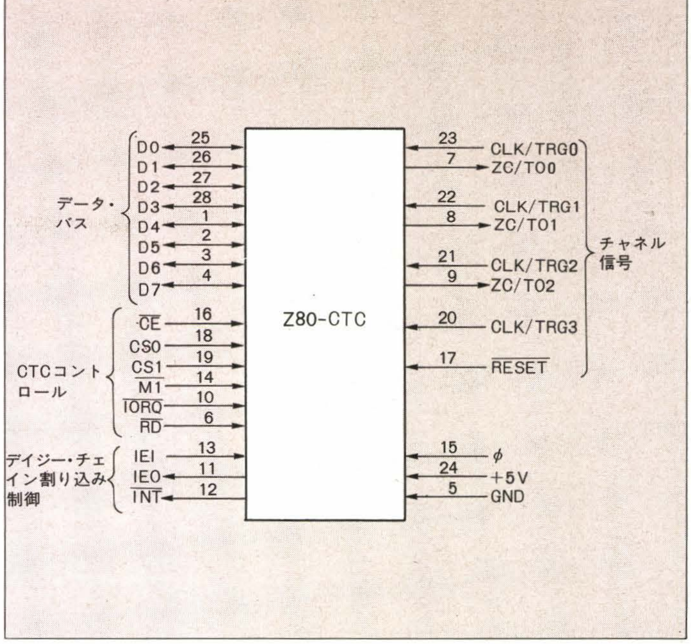


図1

Z80-CTCのピン接続図



このカウンタＩＣとしてＺ－80Ａ・ＣＴＣ（Counter and Timerの略）を使ってみます。Ｚ－80Ａ・ＣＴＣは，Ｚ－80Ａ・ＰＩＯと並んでＺ－80ファミリーの中では低価格で知られていますが，それ以上に使い方も容易であり，分周比をソフトウェアでプログラマブルに設定できますし，４チャンネルもあるので，ボー・レート・クロック作製の用途の他にもさまざまな用途に使うこともできます。

図１にＺ－80Ａ・ＣＴＣのピン配置を示しておきます。さらに図２にＺ－80A・CTCの各ピンの機能を示しておきますから，一応の機能は頭に入れておいてください。

さて，一般によく使われるボー・レートは，75，110，300，600，1200，2400，4800，9600，19200ボーですが，これらの通信速度を保証するためには正確な原発振が必要となります。普通は，これらのボー・レートを正確に保証するために，原発振を 1.8432 ＭＨｚ，4.9152ＭＨｚ，5.0688ＭＨｚ　の整数倍に選ぶのがよいとされていますが，専用の水晶または水晶内蔵型オシレータはまだまだ高価ですので，「Ｘ１」のＣＰＵ ＣＬＯＣＫ周波数である４ＭＨｚを加工して原発振とする工夫をします。このために4ＭＨｚを74ＬＳ163（または74ＬＳ161）

One Point Lesson 1

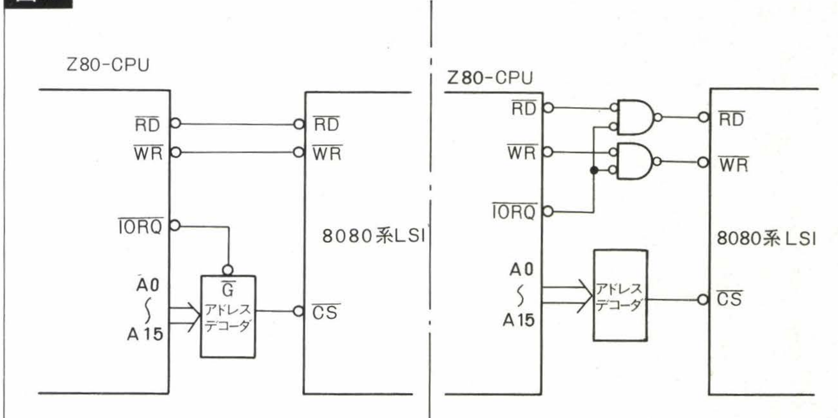
Z-80CPUと8080系LSIの接続例

Z-80CPUと8080系LSI(8251A，8255Aなど)の接続は，Z-80CPUとZ-80系LSI(Z-80ACTCなど）と少々様子が異なることに注意してください。

図Aに誤まったインターフェース例と正しいインターフェース例を示しますから，違いをはっきりさせてください。よく某雑誌なんか(もちろんOh!MZではない！)では，確かめもせずに平気で誤まったインターフェース例を載せていますから。

これは，8080系LSIは$\overline{RD}$や$\overline{WR}$がアクティブ"L"となる前に，$\overline{CS}$がアクティブ"L"になっていなければ動作しないからです。つまり$\overline{CS}$は$\overline{RD}$や$\overline{WR}$に対してセットアップ時間が必要なのです。(a)の誤まったインターフェース例では$\overline{IORQ}$もアドレスデコードに使っていますが，$\overline{IORQ}$は$\overline{RD}$や$\overline{WR}$と同じタイミングで出るために，$\overline{CS}$は$\overline{RD}$や$\overline{WR}$よりもさらに遅れて出力されてしまいます。これでは，セットアップタイムが負になってしまうので動作するはずがありません。

図A



(a)誤まったインターフェース例
(Z-80系LSIであれば正しい)

(b)正しいインターフェース例

図2　Z80-CTCの各ピンの機能

（『マイコンピュータ』Vol.2より）

<table>
<tr><th>ピン名</th><th>ピン番号</th><th>I/O</th><th>信　号　名</th><th>説　　明</th></tr>
<tr><td>D7～D0</td><td>1～4
25～28</td><td>I/O</td><td>Z80-CPU Data Bus</td><td>Z80-CPUとZ80-CTC間のデータの授受を行なうための双方向性バス，D0がLSB。</td></tr>
<tr><td>CS0
CS1</td><td>18
19</td><td>I</td><td>Chip Select</td><td>4つのチャンネルのうちのひとつを選択するためのセレクト・ライン。通常はCPUのアドレス・バスA0およびA1を接続する。
<table>
<tr><th></th><th>CS1</th><th>CS0</th></tr>
<tr><td>チャネル0</td><td>0</td><td>0</td></tr>
<tr><td>チャネル1</td><td>0</td><td>1</td></tr>
<tr><td>チャネル2</td><td>1</td><td>0</td></tr>
<tr><td>チャネル3</td><td>1</td><td>1</td></tr>
</table></td></tr>
<tr><td>$\overline{CE}$</td><td>16</td><td>I</td><td>Chip Enable</td><td>この信号が“L”であるとPIOはイネーブルされる。PIOに対してデータの読み書きを行なうときには，この信号を“L”にする必要がある。通常はアドレス・デコーダの出力を接続する。</td></tr>
<tr><td>ϕ</td><td>15</td><td>I</td><td>System Clock</td><td>Z80-CPUに対するクロックと同じものを用いる。</td></tr>
<tr><td>$\overline{M1}$</td><td>14</td><td>I</td><td>Machine Cycle One</td><td>$\overline{M1}$および$\overline{IORQ}$が“L”のときに，CTCは割り込みベクトルをCPUに対して送る。</td></tr>
<tr><td>$\overline{IORQ}$</td><td>10</td><td>I</td><td>I/O Request</td><td>$\overline{IORQ}$は$\overline{CE}$および$\overline{RD}$と関連して，CPUとCTC間のデータのやりとりを制御する。ライト・サイクルでは$\overline{IORQ}$と$\overline{CE}$が“L”で，$\overline{RD}$が“H”である。$\overline{WR}$はCTCには入力されず。$\overline{RD}$が“H”であることを$\overline{WR}$であると解釈する。リード・サイクルでは$\overline{IORQ}$，$\overline{CE}$および$\overline{RD}$がともに“L”である。
$\overline{IORQ}$および$\overline{M1}$が“L”のときは，割り込み認識シーケンスとなる。</td></tr>
<tr><td>$\overline{RD}$</td><td>6</td><td>I</td><td>READ</td><td>この信号は$\overline{IORQ}$および$\overline{CE}$と関連して，Z80-CPUとPIO間のデータのやりとりを制御する。</td></tr>
<tr><td>IEI</td><td>13</td><td>I</td><td>Interrupt Enable In</td><td>これはZ80の割り込みデイジー・チェイン用の信号であり，この信号が“H”である場合にのみ，PIOは割り込みベクトルを送出することができ，かつRETIシーケンスを認識することができる。</td></tr>
<tr><td>IEO</td><td>11</td><td>O</td><td>Interrupt Enable Out</td><td>これはIEIと関連して，Z80の割り込みデイジー・チェインを制御する信号であり，IEIが“L”であれば，IEOは無条件に“L”となる。またIEIが“H”で当該PIOに割り込み要求があるときは，IEOは“L”となる。</td></tr>
<tr><td>$\overline{INT}$</td><td>12</td><td>I</td><td>Interrupt Request</td><td>CTCの各チャンネルが割り込み可の状態であり，かつダウン・カウンタの値がゼロになったときに，この信号が“L”となりCPUに対して割り込み要求を行なう。オープン・ドレイン型の出力である。</td></tr>
<tr><td>$\overline{RESET}$</td><td>17</td><td>I</td><td>Reset</td><td>この信号が“L”になると，全チャンネルの動作が停止し，チャンネル・コントロール・レジスタの割り込み可ビットはクリアされる。ZC/TOおよび$\overline{INT}$出力はインアクティブ状態となり，IEOはIEIの状態に追随し，データ・バス・ドライバはハイ・インピーダンスになる。</td></tr>
<tr><td>CLK/TRG0
CLK/TRG1
CLK/TRG2
CLK/TRG3</td><td>23
22
21
20</td><td>I</td><td>CLOCK/TRIGGER</td><td>4つのチャンネルおのおのに対してひとつずつCLK/TRGピンがある。カウンタ・モードではこのピンに入力されたパルスのエッジ（立ち上がり，または立ち下がり）により，ダウン・カウンタはトリガされる。タイマ・モードでは，このピンに入力されたパルスのエッジ（立ち上がり，または立ち下がり）によりタイマ動作を起動させることができる。</td></tr>
<tr><td>ZC/TO0
ZC/TO1
ZC/TO2</td><td>7
8
9</td><td>O</td><td>ZERO COUNT/
TIME OUT</td><td>このピンはチャンネル0～2に対してのみ用意され，チャンネル3にはない，カウンタ・モードおよびタイマ・モードの両者において，ダウン・カウンタの値がゼロになったときに，この信号は“H”となる。</td></tr>
</table>

図3　本ボードでのボーレットとCTCのタイムコンスタントレジスタの関係

ボー・レート	①	②	③	④
75	127	31	103	25
110	86*	21*	70	17*
300	31	7	25	6*
600	15	3	12	×
1200	7	1	6*	×
2400	3	×	×	×
4800	1	×	×	×
9600	×	×	×	×
19200	×	×	×	×

①CTCをカウンタモードで用い　SIOのボーレートファクタ＝16
②CTCをカウンタモードで用い　SIOのボーレートファクタ＝64
③CTCをタイマモードで用い　SIOのボーレートファクタ＝16
④CTCをタイマモードで用い　SIOのボーレートファクタ＝64

注1　CTCのカウンタモードはCLK/TRG入力に4MHz/13＝307.69KHzのクロックを入力するものとする

注2　CTCのタイマモードはCLK/TRG入力は開放CPUCLOCK ϕ(＝4MHz)を1/16のプリスケーラモードでカウントするものとする。

注3　＊印はこの値を用いると正確なボーレートは保証されない可能性があることを示す。しかし，低ボーレートでは実際の運用は大丈夫。

注4　×印は不可能であることを示す。

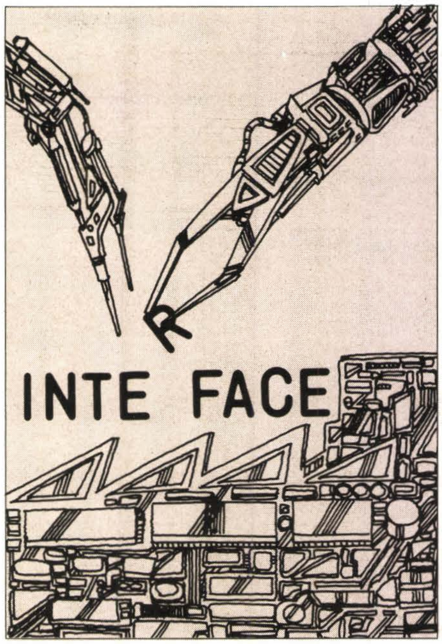
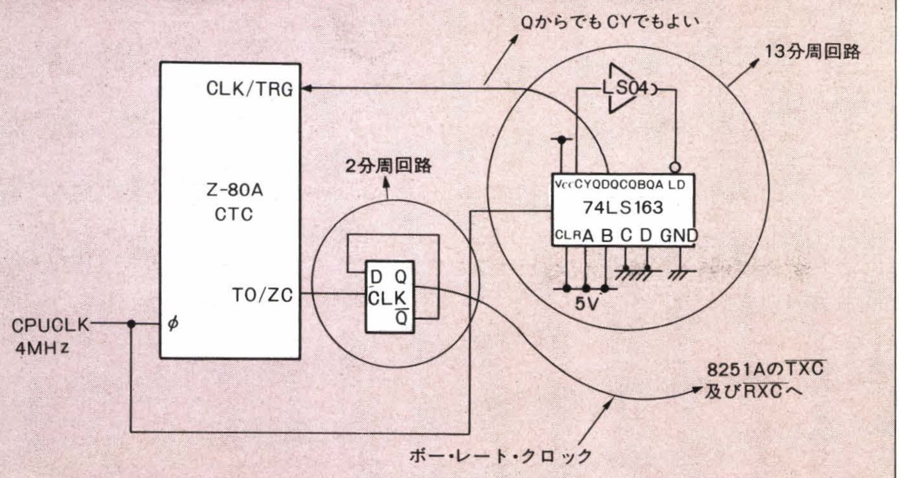


図4

Z-80ACTCにおけるカウンタモード/タイマモードの実際



注：TO/ZCをFlip・FLOPで2分周しているのはTO/ZCのデューティー比が50%を下回わることが予想されるために，それを2分周してデューティー比を50%とするためである。

で13分周した307.69KHzをCTCのCLK／TRG端子に加えることにします。

図3にボー・レートとCTCのタイム・コンスタント・レジスタの値の関係を示します（参考までに，CTCをタイマーモードで使用するときの表も図3にのせておきます。表からわかるようにタイマーモードでは，低いボー・レートでしか正確なボー・レート・クロックが得られません。

以上でCTCによるボー・レート・クロックの作製方法についての説明は終わりです。CTCは，分周比をプログラマブルに変えられるスケーラ(分周器)と考えれば，簡単に理解できますね。

次にSIOである8251Aについて説明します。

タイムコンスタントレジスタへ入力する値の計算例

300ボーでSIOのボー・レート・ファクタが16，CTCをカウンタモードで用いるCTCのCLK／TRG入力周波数は 4MHz／13＝307.69KHz

∴タイムコンスタントレジスタの値をTCとすると次式が成立する必要がある。

FLIP・FLOPで2分周するため

$$307.69\times10^3\,(\mathrm{Hz})\div(TC+1)\div2=300\times16\,(\mathrm{Hz})$$

1足を足していることにも注意　　ボー・レート　　SIOのボー・レート・ファクタ

∴　TC≒31

2．シリアルインターフェースIC(SIO)8251Aについて

8251Aは8080系の周辺LSIでUSART(Universal Synchronous／Asynchronous Receiver／Transmitter）の名称で親しまれていて，シリアルインターフェースICの中でもっともポピュラーな石のひとつです。図5に8251Aのピン配置と，図6に8251Aの各ピンの機能を示します。詳細を述べるといくらあってもページが足りないので割愛しますが，要は8251Aは，送信のときにはデータをパラレルからシリアルに変

One Point Lesson 2

アドレスデコード回路について

「X1」ではI／Oアドレス空間として64Kバイトを使うことができますが，ユーザーに許可されているのは$0～$CFFまでです。本拡張ボードでは，この空間のどの位置にもマッチングできるように，アドレスデコード回路をDIP-SWによってプログラマブルにしています。原理は図Bを見ながら説明していきましょう。

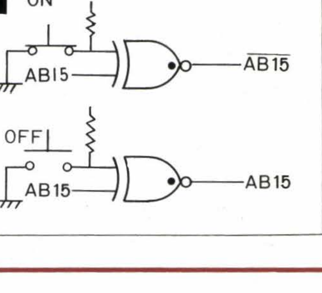
図BでDIP-SWが〝ON″状態のときには，EX-NOR(イクスクルーシブノア)の一方の入力が〝L″となりますので，出力は$\overline{AB_{15}}$（AB_{15}の否定）となります。一方DIP-SWが〝OFF″状態のときには，EX-NORの一方の入力が〝H″となりますので，出力はAB_{15}となります。

このようにEX-NORゲートの一方の入力をSWにして，一方の入力をアドレスとすることによって，出力はアドレスの否定にもなるし，そのままにもなるのです。つまり，アドレスをDIP-SWによって自由に加工することができるわけですから，アドレスデコードをプログラマブルにするのにうってつけであるといえます。

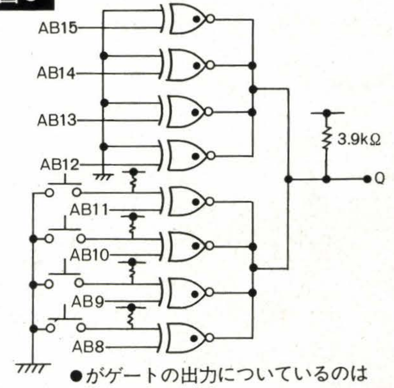
次に，図Cを見てください。これは図CのEX-NORゲートの出力をワイヤードOR接続したものです。

アドレスバスの内容が，EX-NORのもう一方の入力に一致したときのみQ点は，〝H″レベルになります。ですから，アドレス・デコードはオープンコレクタEX-NORをワイヤードOR接続することによって，プログラマブルにすることができるのです。

図B

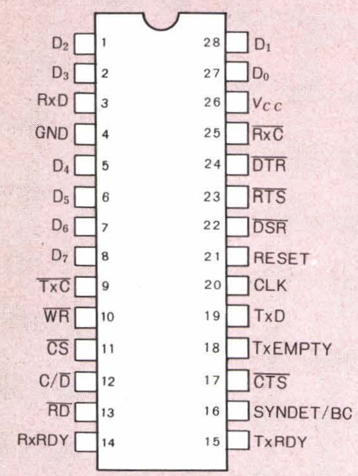


図C



●がゲートの出力についているのは「オープン・コレクタ」

図5 8251Aのピン接続図



換して送り，受信のときには逆にシリアルからパラレルに変換するという作業を，フラグの状態に従って行なっていると考えれば，複雑な機能も簡単そうに思えるようになるでしょう。つまり8215Aは，S→P，P→S変換機能を持ったシフトレジスタのお化けのようなものと考えられます。

3.パラレルインターフェースIC(PPI) 8255Aについて

パラレインターフェースICの名称は，PIO(Parallel Input Output Deviceの略)，PIA (Parallel Interface Adapterの略)とか8255Aに付けられたようなPPI (Parallel Peripheral Interface) などが一般的です。

「X1」はCPUがZ-80Aですから，本来はZ-80A・PIOを使うべきところなの

図7 PIN CONFIGURATION

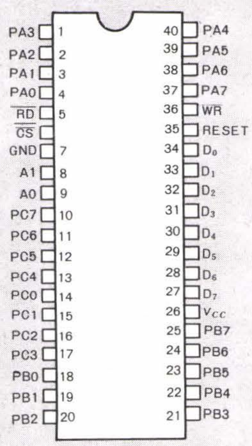


図6 8251Aの各ピンの機能

種類	ピン番号	ピン名	信号名	I/O	機能
データ・バス	1，2 5～8 27，28	D_7～D_0	データ・バス	I/O	8251Aはこの8本の双方向性データ・バスを通じてCPUとデータ，モード・ワード，コマンド・ワード，ならびにステータス・ワードのやりとりをする。
	21	RESET	リセット	I	この信号が"H"になると8251Aはリセットされ，アイドル状態になる。この状態ではモード・ワードを受け取るシーケンスになる。この信号のハルス幅は最低6クロック・サイクル(ピン20)である。もちろんこの場合クロックは動いている必要がある。
	20	CLK	クロック	I	この信号は8251A内部のタイミング・コントロールに使用される。通常は，8224の$\phi 2$または8085AのCLKOUTが用いられる。このクロックは，シリアル・データのタイミングとは無関係であるが，レシーバまたはトランスミッタ・クロック(RxC，TxC)の30倍以上の周波数でなければならない。
	10	$\overline{WR}$	Write	I	この信号が"L"であると，データ・バス上のデータが送信データ，モード・ワード，またはコマンド・ワードとして8251A内部に書き込まれる。8080Aシステムの場合には$\overline{IOW}$が用いられ，8085Aシステムの場合にはIO/$\overline{M}$・$\overline{WR}$が用いられる。
	13	$\overline{RD}$	Read	I	この信号が"L"であると，受信データまたはステータス・ワードがデータ・バス上に乗せられる。8080Aシステムでは$\overline{IOR}$が用いられ，8085Aシステムでは IO/$\overline{M}$・$\overline{RD}$が用いられる。
	12	C/$\overline{D}$	Control/Data	I	この信号によりリード／ライトの対象がデータであるのかコントロール・ワードであるのか指定する。通常はアドレス・バスのA_0が接続される。
	11	$\overline{CS}$	Chip Select	I	この信号が"L"であることにより，8251Aをセレクトする。この信号が"H"であるときには，$\overline{RD}$および$\overline{WR}$は無効である。
モデム・コントロール	22	$\overline{DSR}$	Data Set Ready	I	モデム・インターフェースではデータ・セット・レディとして用いるが，モデムを用いないときには何の目的に利用してもよい。
	24	$\overline{DTR}$	Data Terminal Ready	O	モデム・インターフェースではデータ・ターミナル・レディとして用いるがモデムを用いないときには何の目的に利用してもよい。
	17	$\overline{CTS}$=L	Clear to Send	I	モデム・インターフェースではクリヤ・ツー・センドとして用いる。この信号が"L"であるときのみ，トランスミッタはイネーブルされる。
	23	$\overline{RTS}$	Request to Send	O	モデム・インターフェースではリクエスト・ツー・センドとして用いるが，モデムを用いないときには何の目的に利用してもよい。
トランスミッタ	19	TxD	Transmit Data	O	このピンを通じて送信データが出力される。トランスミッタがアイドル状態のときにはマーク状態("1")となる。
	15	TxRDY	Transmitter Ready	O	この信号が"H"であると，トランスミッタが次のデータを送信する準備ができていることを示す。データをライトするとリセット("L")される。
	18	TxE	Transmitter Empty	O	この信号が"H"であると，トランスミッタのバッファ・レジスタが完全に空であることを示す。データをライトするとリセットされる。
	9	$\overline{TxC}$	Transmitter Clock	I	この信号の立ち下がりにより，トランスミッタのシフトレジスタは送信データをシフトする。
レシーバ	3	RxD	Receive Data	I	このピンを通じて，シリアル・データは8251Aに入力される。
	14	RxRDY	Receiver Ready	O	この信号が"H"であると，RxDから入力されたシリアル・データをパラレル・データに変換し終わり，CPUに対してデータを渡す用意ができていることを示す。このデータをリードすることにより，この信号はリセットされる("L")。
	25	$\overline{RxC}$	Receiver Clock	I	受信データを再生するためのクロック，この信号の立ち上がり時にデータをサンプルする。
	16	SYNDET /BRKDET	Sync Detect /Break Detect	I/O	同期式モードでは出力(内部Sync検出モード)，または入力(外部Sync検出モード)となり，非同期式モードでは出力となる。 ●同期式モード (SYNDET) 内部Sync検出モードでは8251AがSyncキャラクタを検出したときに"H"となる(出力)，この信号はステータス・ワードをリードすることにより，リセットされる。 外部Sync検出モードでは入力信号となり，この信号の立ち上がりにより，8251Aに対して同期がとれたことを示す。 ●非同期式モード(BRKDET) 非同期式モードではこの信号は出力のみとなり，8251Aがブレーク状態を検出したことを示す("H")。ブレーク状態というのは，2文字分に相当する期間RxDの状態がスペース("L")であったこと。

でしょうが，Z-80A・PIOや68系のPIAである6821などでは，ポートのビット数が16ビット(16本の入出力線があるということ)であるのに対して，8255Aでは24ビットもの入出力線を有しているので，ROM WRITERをこのPPIに継ぐ場合には8255Aのほうが有利であると判断してＰＰＩとして8255Aを採用しました。

8255Aのピン配置のみ，図7に示しておきます。

各ピンの機能に関しては，来月号以降に製作予定のP-ROM WRITER (MOSのROM，たとえば2716，2732，2764ばかりではなく，TTLのFUSE・ROMライタもカバーします）の欄で詳しく説明します。

4.SIOとCTCの動作のチェック法

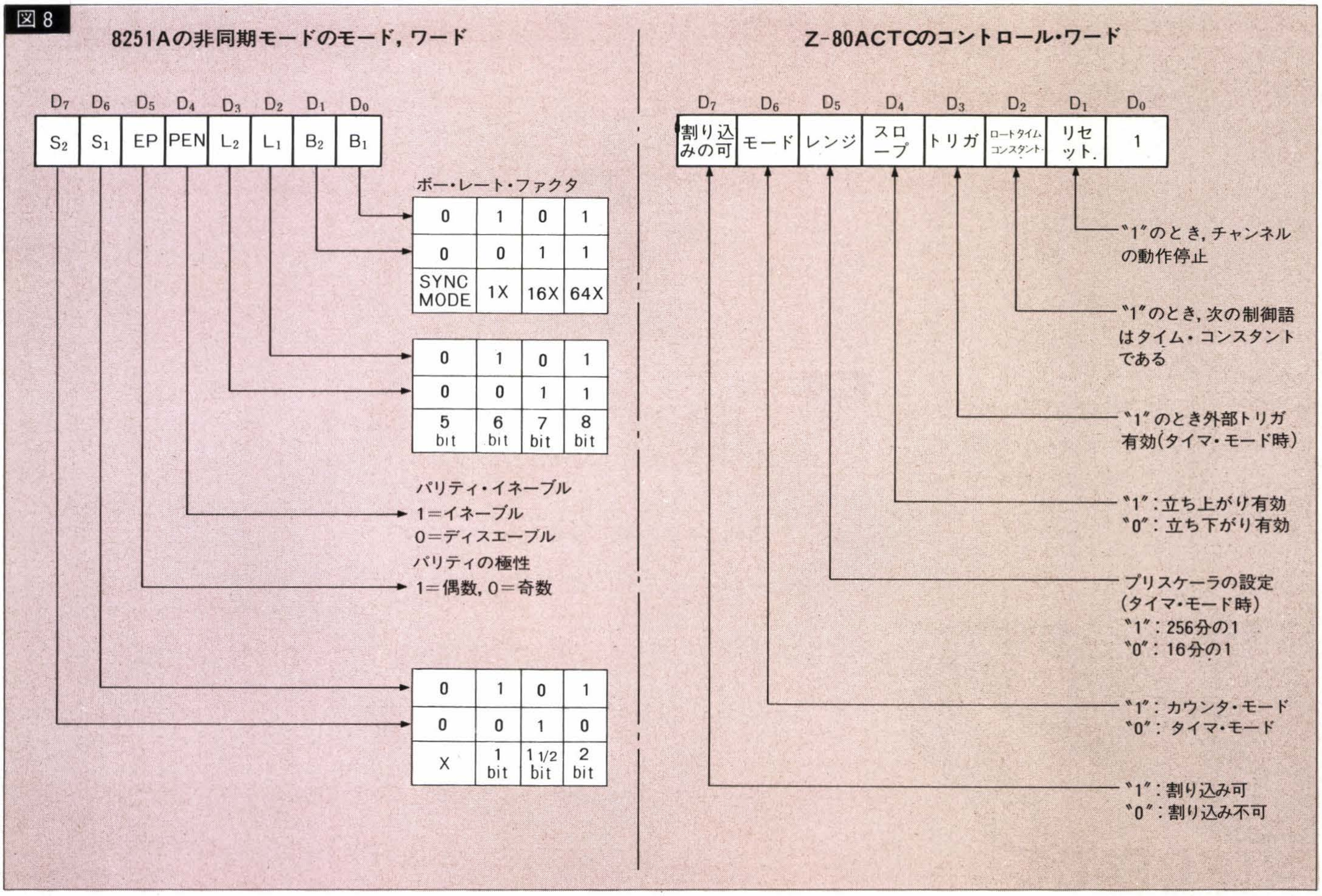
まず，ＣＴＣ周辺の配線をチェックしてください。オシロスコープのない場合には，クロックが発生しているかどうかを目でみることが困難ですので。

次にＳＩＯの動作が正常であるかどうか

リスト1

```
5   CT=&HC00:SD=&HC04:SC=&HC05
10  '  Loop Back Test for 8251A
20  '  Initial Set for Z-80A CTC
30 OUT CT,&H47                 'Set Counter Mode
40 OUT CT,31                   'Baud Rate Factor=16,Baud Rate=300
50  '  Inital Set for 8251A
60 OUT SC,0:OUT SC,0:OUT SC,0 'Make Sure that USART is in Command Word Sequence
65 OUT SC,&H40                 'Inital Reset
70 OUT SC,&H7A                 'Mode Word to SIO
80 OUT SC,&H15                 'Error Reset & TX,RX ENABLE
90 A$=INKEY$
100 IF INP(SC) AND 1=0 THEN 100'Check for Transmitter Ready
110 OUT SD,ASC(A$)
120 IF INP(SC) AND 2=0 THEN 120'Check for Receiver Ready
130 PRINT CHR$(INP(SD));       'Print Input Character
140 GOTO 90
150 END
```

図8



をポーリング・モードによるループ・バック・テストで行ないます。ループ・バックというのは背合わせ接続のことで，8251AのTXDとRXDを接続することです。リスト1にそのためのプログラムを示します。

このプログラムは，キーボードから押された文字を画面上に表示するだけですが，これが正常に動けば，8251Aの動作はほぼ正常であるといえます。

図8～10に8251AとZ-80ACTCのコントロール方法を示しておきますので30，40，70行のパラメータを変えて実験してみてください。

おわりに

今月は「X1」に「シリアル・パラレル・カウンタ」インターフェースをつけてみました。周辺ICの代表的なものをすべて付けた少々欲ばった設計となりましたが，市販されているものを買うより格段に安く作れますので，「X1」を単なるゲーム・マシンと考えたくない人はハードウェア製作実践の第一歩として本ボードを作ってみたらいかがでしょう。

来月は，RS-232Cによるターミナルプログラムか，8255Aの応用例ROM WRITERの製作のいずれかを掲載します。ご期待ください。

参考文献

1) マイコンピュータ No.2
2) トランジスタ技術 1982，11月号
(以上，CQ出版)

付録

RS-232Cインターフェースでは，±12Vの電源が必要であると，先月号でも書きましたが，「X1」の拡張I/Oポートには±12Vの電源が出ていません。外部から供給してもいいのですが，あいにく電源の持ち合わせがないという人のために，5Vから±12Vの電圧を得る回数を示しておきますから作ってみてください。

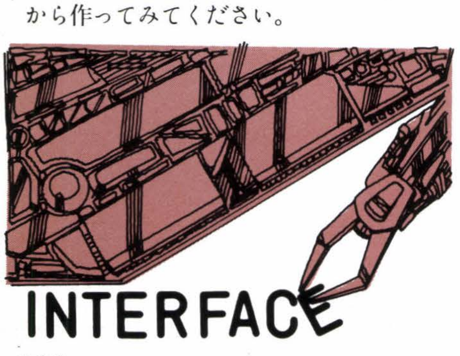


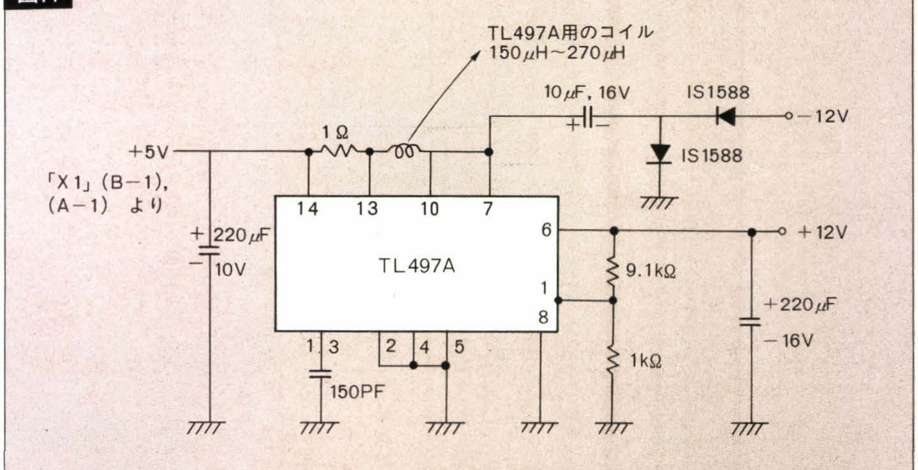
図9 本ボードでのI/Oアドレス・マッピング
(SW0=SW1="OFF", SW2=SW3="ON"とすること)

$C00	CTC	チャンネル0 コントロール及びデータ・レジスタ
$C01	CTC	チャンネル1 コントロール及びデータ・レジスタ
$C02	CTC	チャンネル2 コントロール及びデータ・レジスタ
$C03	CTC	チャンネル3 コントロール及びデータ・レジスタ
$C04	SIO	データ・レジスタ
$C05	SIO	コントロール・レジスタ
$C08	PPI	Aポートデータ・レジスタ
$C09	PPI	Bポートデータ・レジスタ
$C0A	PPI	Cポートデータ・デジタル
$C0B	PPI	コントロール・レジスタ

図10 パーツリスト

74LS266×2　抵抗3.9kΩ×5 74LS02×1　電解コンデンサ22μF 74LS04×1（電源とGNDの間に入れておく） 74LS32×1 74LS163×1 74LS138×1 74LS74×1 74LS245×1 Z-80ACTC 8251A 8255A MC1488 MC1489 DB-25S(RS-232C用のメス型コネクタ1)「X1」用ユニバーサルボード(サンハヤトまたはI/Oデータ機器) DIP スイッチ×1
±12Vをボード内の作る人のために
TL497A 1Ω (1W) 1kΩ (1/4W) 9.1kΩ (1/4W) 150μH～270μHのコイル 220μFの電解コンデンサ×2 150PFコンデンサ 10μF電解コンデンサ ダイオード 1S1588×2 (総額 約8,000円)

図11



※注：1Ωの抵抗には必ず1Wの大型の抵抗を使ってください。

図12

1. 拡張I/Oポート(オプション)

メイン基板コネクタ

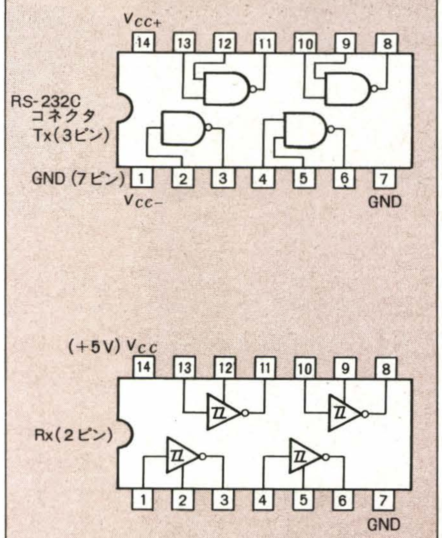
端子	B	A
1	+5V	+5V
2	DB3	DB2
3	DB4	DB1
4	DB5	DB0
5	DB6	GND
6	DB7	AB15
7	CPU CLK	AB14
8	$\overline{M1}$	AB13
9	$\overline{WR}$	AB12
10	$\overline{RD}$	AB11
11	$\overline{IORQ}$	AB10
12	$\overline{MREQ}$	AB9
13	GND	AB8
14	$\overline{HALT}$	AB7
15	IEI	AB6
16	アキ	AB5
17	RESET	AB4
18	$\overline{EX\ IO}$	AB3
19	$\overline{EX\ INT}$	AB2
20	$\overline{EX\ WAIT}$	AB1
21	$\overline{NMI}$	AB0
22	GND	GND

2. 拡張I/Oポート端子表

各スロット(1ー4)コネクタ

端子	B(半田面)	A(部品面)
1	+5V	+5V
2	DB3	DB2
3	DB4	DB1
4	DB5	DB0
5	DB6	GND
6	DB7	AB15
7	CPU CLK	AB14
8	$\overline{M1}$	AB13
9	$\overline{WR}$	AB12
10	$\overline{RD}$	AB11
11	$\overline{IORQ}$	AB10
12	$\overline{MREQ}$	AB9
13	GND	AB8
14	$\overline{HALT}$	AB7
15	IEI(1-4)	AB6
16	IEO(1-4)	AB5
17	RESET	AB4
18	$\overline{EX\ IO}$	AB3
19	$\overline{EX\ INT}$	AB2
20	$\overline{EX\ WAIT}$	AB1
21	$\overline{NMI}$	AB0
22	GND	GND

図13 SN75188N



One Point Lesson 3

データ・バスのバッファリングについて

アドレスは，DMA コントローラを使う場合を除いては単方向性ですので，バッファリングに注意する必要はあまりないのですが，データバスは双方向の共通バス方式なので，バッファリングには十分注意する必要があります。通常データバスのバッファリングには74LS245とか74LS 640 (74LS 245の反転版) を使うのですが，バッファリングは$\overline{G}$ (ゲート入力，74LS245の19番ピン) を必要に応じて (アドレスコード回路からの出力に応じて) "L" に落とすのです。

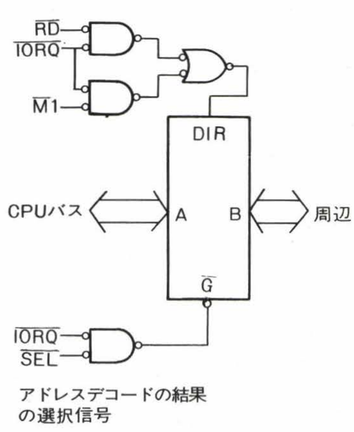
もうひとつ別の考え方もあります。それは$\overline{G}$="L"に保っておき，DIR端子(Direction端子で74LS245では1番ピン) を必要に応じて切り換えるという方法です。図Dを見てください。

(a)のほうは，$\overline{G}$を選択信号がアクティブであり，かつI/O に対するアクセスであるときアクティブにしようとしています。これは，自分がアクセスされてないときにはCPUバスと周辺バスを切り放すことを前提とした考えて，至極当たり前の回路です。

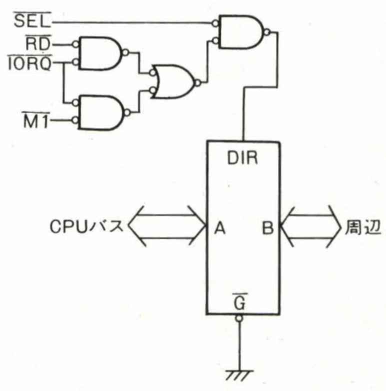
(b)のほうは，自分が選択されていてかつ，読み込みサイクルであるか，割り込み応答サイクルのとき，B→AのSWを"ON"としてA→BのSWを"OFF" としています。それ以外のときは，バスの方向はA→Bとなります。

(a)の回路ですと，Z-80が"RETI" (return from interrupt) を実行したのを周辺側に認識させることができないのですが，(b)の回路ですと，自分が選択されていなくてもバスがA→BつまりCPUのバスが周辺へと継ながっているので，"RETI" の実行を認識でき，MODE2の割り込みの終了を認識することができます。MODE 2の割り込みを使いたい場合には(b)の回路を使ったほうがベターですが，精神衛生上からは(a)の回路のほうがいいといえます。

図D



(a) 通常考えられるバッファリング回路



(b) Z-80 MODE2
割り込みを意識したバッファリング回路

MZ システムソフトの活用法 3　　本格的マシン語開発ツール

FLOPPY DOS

風間　浩

前回・前々回と，テープベースのマシン語関連のソフトを紹介してきましたが，いよいよ今月はフロッピーディスクを使った本格的なマシン語開発ツール「Floppy DOS」についてお話ししたいと思います。

このFloppy DOSというのは，かなり大規模なシステムですから，1回では解説し終わることができません。ですから，まず今月は全体的な構造から見ていきたいと思います。

表1　MZ-80Bでの対応名称

MZ-2000/2200	MZ-80B/B2
Floppy DOS	F-DOS
MZ-2Z004	SB-7201
2Z-004A	SB-1521
MZ-1Z001M	SB-1520
MZ-2Z002	(ありません。SB-6521+カラーグラフィックと考えられます)
MZ-1Z005	MZ-8BT04 (SB-2x01 シリーズ)

DOSって何だろう

DOSというのは，Disk Operating System の頭文字を取った略語で，「(フロッピー) ディスク装置を操作するために使うソフト」という意味で使われています。システム，というのは，普通は機械（ハード）まで含めた言い方が多いのですが，ここではソフトだけを指します。

マシン語に興味のある皆さんなら，「モニタ」を使ったことがあるはずです。MZ-1200 (80K/C/K2/K2Eを含みます。以下同じ）に入っているSP-1002はやや機能不足ですが，MZ-2000(2200も含みます。80B/B2も同様ですが，表1の用語は読み換えてください）に入っているMZ-1Z001Mでは，次のような働きを持っています。

○キーボードから文字を受け付けること
○ディスプレイに文字を表示すること
○カセットテープにSAVEしたり，LOADしたりすること
○音楽を演奏すること
○コマンドを受け付けること。また，その処理をすること

大まかに見ればこんなところでしょう。アセンブルリストが公開されていますので，読んだことのある人ならある程度は知っていることも多いかと思います。

MZ-1200でも，この中の「コマンドを…」の部分が弱い（すなわちコマンドが少ない）だけですので，考え方は同じです。

ここで，「モニタ」の対象となっているデバイス（CPU・メインメモリ以外の装置）に注目してください。それを列挙するなら

○キーボード
○ディスプレイ（文字画面のみ）
○スピーカー
○カセットデッキ（CMT）

だけです。

これがBASIC MZ-2Z001になりますと，次のデバイスが追加されています。

○プリンタ
○グラフィック
○RS-232C ｝MZ-80BのSB-6520や，テープ版（MZ-2000のMZ-1Z010を除く）のBASICにはありません。
○GP-IB
○フロッピーディスク

ということは，

○モニタのDコマンドを，プリンタに対して行ないたい。
○同じくSコマンド，Lコマンドをディスクに対して行ないたい。

のような希望は残念ながらかなえられません。

つまり，モニタは明らかに現在の豊富なデバイス・周辺装置に遅れを取っているわけです。しかもこれは，現在の話であって，将来的に見て新たなデバイス「ハードディスク，標準(8インチ)・マイクロ（3インチ）ディスク，バブルカセット，音声合成，音声認識，ライトペン，映像入力，ジョイスティックなど，簡単に予測できる範囲でもこれだけある」が使用可能となった場合にも，それに対応させることができません。これはBASICにしても同じで，新しいデバイスができたなら即，新命令が作られるということが多いでしょう。

また，新命令といえばBASICではもちろんのこと，モニタでさえ追加することは（不特定多数個あるなら，なおさら)困難です。モニタに「サーチ機能」や「転送機能」を追加する，なんていうのは難かしいわけです。

話がかなり広範囲に及んでしまったようですが，これらをまとめると，

○モニタはデバイスを増やせない。特にディスクが扱えない。
○モニタはコマンドを追加できない。

の2点に絞ることができます。

DOSというのは，この2点を解決するように「モニタを拡張した」内容と考えてよいでしょう。すなわち，

○ディスクをそのまま使えるようにした。

○デバイスが追加できるようにした。

○命令を追加できるようにした。

ということです。

このうち，「ディスクを…」については，単なるLOAD/SAVEをディスクに対して行なえるだけの拡張にとどまっていません。ディスクBASICでお馴じみのRENAME（ファイルネームのつけかえ），DELETE（ファイルの削除）なども含まれているのです。つまり，BASICで行なっていたことの一部も吸収しているわけです。また，コマンドの名前が1文字でなく，RENAMEのようなわかりやすいものになっています。そうでなくてはコマンドも26個までしか増やせませんしね。

一方，コマンドの拡張については，「増やしすぎるとメモリが足りなくなる」というマイナス面が出てきます。ですから，次のような方法がとられました。

○新命令（はじめから入っていない命令）が使われたなら，それをファイルネームとして「ディスク中」から探し出し，自動的にロードして実行に移る。

DOSには必要最小限の命令だけを備えておくことにして，新命令はディスクにおきとどめれば，メモリを食い過ぎることがない，というわけです。ディスクは280Kバイト（MZ-2000のミニフロッピーディスクの場合）もの容量があるのですから，かなり自由な命令が追加できることがわかりますね。

DOSにはどんなものがあるか

現在一般のパーソナルコンピュータで普及しているDOSには，

CP/M-80（8080/Z-80用）（米国デジタルリサーチ社製）

CP/M-86（8086/8088用）（同上）

MS-DOS（同上）（米国マイクロソフト社製）

H-DOS（Z-80用）（ハドソン製）

Floppy DOS（同上）（シャープ製）

などがあります。

MZ-2000では，これらのうちCP/M-80，Floppy DOSが使えます。

MZ-1200では，H-DOS，Floppy DOS（商品名はF-DOS），それにハードの追加によりCP/M-80も何とか動きます。

CP/M-86とMS-DOSは，8086/8088といった16ビットCPU用ですから，MZのうち3500以下の型番の機種では動きません。が，MZ-2000/2200に16ビットボードを取り付けている場合は可能性としては残されています。

H-DOSのMZ-2000用は，筆者は個人的には移植することに成功しています（市販はされていません）。

DOSは何のために使うか

その昔，「パーソナルコンピュータ」という言葉がほとんど使われていなかったころ，「マイクロコンピュータ(マイコン)」を自作する人が多くおりました。彼らは，ハードウェアが好きな人たちですから，CPUやRAMなどを部品で買ってきて，ハンダづけをして作っていました。CPUは8080や，その完全上位コンパチブルなZ-80が大半を占めていました。

ところが，知ってのとおり「マイコンはソフト無ければただの箱」ですので，ソフトまで自作せざるを得ません。しかし，ハードが得意な人はソフトは苦手，ソフトが得意な人はハードは苦手であることが多い，という状況は今も昔も変わりません。よって，マイコンを自作した人の多くは，できるだけ少ない労力でソフトができたら…と考えることになります。

さきほど出てきた「CP/M」というDOSは，こんな人たちに多く受け入れられました。また，できるだけ多くの人に利用してもらえるよう，互換性を最重要視し，次の点が配慮されています。

○CPUは8080でもZ-80でもよいように，8080の命令しか使わない。すなわち，Z-80特有の高性能命令はすべて使わない。これは，CP/M本体もCP/Mの下で動くソフトにも共通です。

○カタカナが扱えないマシンもあるかも知れないので，キャラクタコード7FH以下のもの（アルファベット・数字程度）しか使わない。

○グラフィックが扱えないマシンもあるかも知れないので，グラフィックは使わない。

○カラーが扱えないマシンもあるかも知れないので，カラーは使わない。

○ディスクを使わないとDOSでなくなるので，片面単密度の標準（8インチ）ディスクだけが使える。ミニフロッピーやカセットは使わない。

○ターミナル方式のマシンもあるかも知れないので，スクリーンエディットは使わない。

○RAMは0番地から入っていればよい。

というものです。つまり，8インチのディスクと0番地からのRAMがある，という大前提を作り，その他の部分は最小限度のギリギリのところまで性能を落としているわけです。CP/Mを購入したユーザーは，自分のマシンのハードに合わせ，ごく一部の改造を行なえば使えるようになります。

CP/Mの発祥の地であるアメリカでは，日常からタイプライタに慣れた生活をしているため「タイプミス」もほとんどなく，スクリーンエディットなしのアルファベット・数字のみが，結構すんなりと受け入れられました。そして，マイクロソフト社がBASIC（もちろんグラフィックなどは何もない），BASICコンパイラやFORTRANなどの言語をいくつか供給し，英文ワードプロセッサなどのソフトも多く生まれました。

一方，日本では自作派の人が（アメリカに比べ遅かったため）多くなる前に，MZ-80K（1978年）やPC-8001（1979年）などのパソコンと呼ばれる機種が増えたのです。これらは0番地からのRAMはなく，ROMが占めています。したがってCP/Mを動かすには，外付けハードが必要でした。またフロッピーディスクは5インチです。このためユーザーが移植作業をするのは事実上困難（第一，8インチのディスクを読み出すことすらできない）ですから，輸入業者がまとめて大改造と移植とを行ないました。MZ-80Kのディスクは5インチ両面単密度，PC-8001は同片面倍密度性で，互換性

はありません。これはハードが違うため当然でしたが，問題はこの後です。MZ-80Bが両面倍密度となり，FM-8，PC-8001（ディスクの仕様が変更された），PASOPIAと続き，以前から両面倍密度だったIF-800モデル20と合わせ，この規格で5インチのミニフロッピーディスクについてのハードだけは統一されました。にもかかわらず，CP/Mの移植はバラバラに行なわれたため，本来の重要な目的のひとつであったはずの「互換性」はことごとく失なわれました。したがって，数多くあったソフトも，言語などごく一部が使える程度で，日本で作られた日本人に使えるソフトはほとんどといってよいほど，発表されませんでした。8インチのディスも，現在は両面倍密度が常識ですから，やはり大がかりな移植作業が輸入業者等の手によって再び各社バラバラに行なわれました。これにより8インチでの互換性も失なわれたわけです。

本当は，CP／MのもとでBASICも動くのですから，これが普及しても不思議はないのですが，（他機種より）優れた部分が一切サポートされない（グラフィック命令など，ただのひとつもない）ため，やはり各社ともバラバラにBASICを供給しました。もちろんCP／Mとは関係なしに。

これで，CP／M用にBASICコンパイラがあるのに，いつも使っているプログラムをそのままコンパイルすることもできなくなったのです。結局のところCP／Mは，アセンブリ言語でマシン語プログラムを作りたい人が使うことがほとんどです。

この状況に前後して，日本製のDOSも作られました。これらのDOSは，まずどの機種にも通用するソフトはそのマシンの性能を最低限で使用するに過ぎない，という点に特に注目し，BASICのようにその機種の性能をフルに生かしたものが必要だ，という思想のもとに作られました。パソコンにはパソコンのDOSがあってもいいではないか，というわけです。

そのうちのひとつがH-DOS，もうひとつはF(loppy)-DOSです。ともに，MZでのみ動きます。ハドソンにしても，シャープにしても，BASICそのものを作るだけの技術力があったからこそ，独自のDOSを作れたのです。外国にBASICを注文している他のメーカーには，MZのようなマネはできません。MZはマシン語に強い，と昔からいわれている原因のひとつはこの辺にあるようです。

この結果，MZでF-DOSを使えば，アセンブリ言語を使わずにBASICコンパイラでマシン語プログラムを作ることができるようになりました。私は，Floppy DOSこそ，MZにとって最大の武器であると考えています。

Floppy DOSの特長

さきほど，Floppy DOSはMZのハードをフルに生かしている，といいました。この部分を詳しく解説してみましょう。

◎Z-80の命令をフルに利用している。

CP／Mやマイクロソフト BASICは，8080の命令だけで作られていました。マイクロソフトBASICを拡張したBASICもまた，8080の命令がほとんどですから，メモリ効率・スピードが悪かったのですが，シャープのBASICやHuBASICはZ-80を生かしたものになっているため，速くてメモリ効率もよくなっていました。

DOS本体も同じことで，スピードは（もともとディスクを扱うソフトなので）大差ないにしても，メモリ効率はIYレジスタ，裏レジスタの活用によりかなりアップしています。

◎はじめから高水準のソフトが付属している

CP／Mには，ラインエディタ（スクリーンエディットできないエディタ），8080アセンブラ（8080の命令のみ。マクロ機能なし。直接マシン語を発生してしまう）が付属ですが，MZ-2000のFloppy DOSでは，スクリーンエディタ，Z-80マクロアセンブラ(Z-80の命令がすべて使える。マクロ機能もある。RBファイルを作成するため，ライブラリ化，多数をリンクすることもできる)，実数型BASICコンパイラ，整数型BASICコンパイラが付いています。

CP／Mでこれらを行なうには，エディタとアセンブラだけでも

○CP／M-80　65,000円…DOS
○EDIT-80　36,000円…スクリーンエディタ
○MACRO-80　60,000円…マクロアセンブラ

が必要で，合計161,000円です。BASICコンパイラは120,000円ですから，合わせると「ディスクとプリンタを買ってもまだおつりがくる」ほど高額です。しかも高速化された整数型のコンパイラはありません。BASICコンパイラでグラフィック命令を使うことも不可能です。

これに対してFloppy DOSは，50,000円しかしません。コストパフォーマンスは非常に高いといえましょう。

MZ-1200のF-DOSは，エディタ（ラインエディタ），アセンブラ（マクロはないがZ-80の命令はすべて使うことができますし，RBを作成する方式です）で20,000円です。CP／Mの（改造用ハードを含めた価格）85,000円と比べてみてください。BASICコンパイラはオプションですが，わずか10,000円です。私は別にシャープのまわし者(？)ではないのですが，良心的なこの値段設定にはただただ脱帽したいと思います。

◎アドレスを意識せずに開発できる

RBを介する方式を採用したことで，原始ファイル（ソースリスト）の入力時からデバッグに至るまで，何番地，といったアドレスを意識することなく行なうことができます。

◎数多くのデバイスが利用できる

標準でディスク4台以外に5個のデバイスがサポートされ，8個のデバイスがオプションで，さらに7個を自由にユーザーが定義して使えます。

この他にも数え切れないほどの特長が備えられていますが，その詳細な解説は，その部分が出てきたときにいたします。

どのような目的で使えばよいのか

まず，前回の「システムプログラム」のディスク版として使うことができます。

MZのBASICやFloppy DOS，パスカルなど，ほとんどのソフトはFloppy DOSによって作られているとされています。もう少し詳しく説明すれば，

① ミニコン（恐らくNOVAだと思う）でSP-1001・SP-1002（モニタ）とSP-5002（MZ-80K用BASIC）を作った。
② 同様にしてシステムプログラムを作った。
③ ①を②に打ち込み直した。つまり，SP-5002のソースリスト

を，MZ-80Kのシステムプログラム上に手で入力し直した。

④　SP-5002を改良してSP-5010，SP-5020，SP-5030，SP-6010（ディスクBASIC）を作り順次発売した。

⑤　②を使ってF-DOS（SP-7010）を開発した。

⑥　⑤を使ってパスカル（SP-4010）を開発・発売した。

⑦　⑤のデバックを終了させ，発売に踏み切った。

⑧　⑤を使ってMZ-80B用各種BASIC，システムプログラム，F-DOS（SB-7020）を移植・発売した。

⑨　F-DOS（SB-7020)に，それまでに開発したソフトのソースリストを入力（テープを介したはず）した。

⑩　⑨を使って，MZ-2000用を開発し，発売した。

のようであったと考えられます。各ソフトの内部構造や発売時期からみても，多少の前後はあり得ますが，大まかにはこのようであるはずです。

すなわちFloppy DOSは，MZで使うためのソフトなら大規模なものまで自由に開発することができる，と考えてよいでしょう。アマチュアからプロまで，どんな人にも使えるツールなのです。考え方を変えれば，Floppy DOSを逆アセンブルしたうえでソースリスト化を図り，自由に改造したり，新しいBASICを作ったり，新しい言語（シャープのパスカルのように）を作ったりも，少なくとも「可能性」としては持っていると，といえます。これほどのシステムが「それ自身」を作ることができたり，ましてやそれが一般のユーザーにも開放されている，なんてことは，まず滅多にあることではありません。皆さんは非常に恵まれた環境にいるわけです。

また別な使い方としては，「自分にはマシン語ができないけれど，速いプログラムが作りたい」という人のために，「BASICコンパイラ」だけを利用する，ということも考えられます。BASICコンパイラは，BASICで書かれたプログラムをマシン語に変換するソフトで，ややムダがでるため「はじめからマシン語で作った（アセンブリ言語を使ったプログラム」よりは遅いのですが，「BASICインタープリタでの実行速度」よりはずっと速くなっています。マシン語は何にも…という人には，これだけを使うのもFloppy DOSの有効的な利用法だということを付け加えておきましょう。

F-DOSとH-DOSとの関係（MZ-1200のみ）

MZ-80Bが登場する直前——MZ-80K2E，1200がまだ発表されておらず，MZ-80K，C，K2だけがあったころです——シャープ初のDOS，F-DOSが発売になりました。1981年のはじめです。ちょうどPASCALやHuBASICがデビューしたり，CP／Mが使えるようになったり（あまり話題にはなりませんでしたが），ハード・ソフトも詳細に公開されたりして，MZが極めて使いやすくなった時期でした。

ところが，MZでは既にDOSと呼ばれるものが走っていたのです。その名は「H-DOS」。MZ-80K/Cのユーザーなら知らない人はいないでしょう。ここで当然のことのように，「H-DOS vs F-DOS」という争い(？)が生まれることになりました。でも，この軍配は今だどちらにも上がっていないようです。それは，それぞれに大きな特長があるからです。

F-DOSの特長は

○実アドレスを意識せずに開発できる。

○BACICコンパイラが使える（10,000円のオプション）

○デバイスが豊富（特にカセットが使える）

などがあります。これらは既に説明済みですね。

対するH-DOSは，

○フリーエリアが広い。

○アセンブラにCHAIN疑似命令がある。

○スクリーンエディタが使える。

○ディスクBASIC（SP-6010）とディスケットが共用できる。

○トレーサー（デバッガ）が使える。

などがあります。知ってのとおり，RAMは最大でも48Kバイトしかないため，「フリーエリアが広い」という点，そしてCHAINがある点が大規模なプログラム作りの際に高く評価されました。ハドソン自身もHuBASIC V1.0～1.3とX1，MZ-700用の同V2.0をH-DOSによって開発しています。X1用HuBASICは40Kバイトですから，F-DOSでは不可能（CP/Mでも不可能）なことなのです。

スクリーンエディタや，ディスクBASICとのファイル共存も便利ですし，ディスクBASICでは不可能なマシン語ファイルの直接（ディスクへの）SAVE，ディスクの中身をセクタ単位で読み出したり変更すること，各ファイルのディテール（ファイル長，アドレス，その他）を見ることができるなども可能です。

H-DOSはちょっとしたマシン語プログラムと，超・大規模なプログラム開発用にはかなり向いていることがわかります。

F-DOSはこれと比べると，やはり何といってもBASICコンパイラが最大の特長でしょう。アドレスをも見のがすことはできません。

やはりここでも決着はつきませんね。ただ，H-DOSのほうは別の機会に，ということにしておき，今回はF-DOSのほうをしっかりと勉強（かな？）することにしましょう。今後MZ-2200などを使うことがあれば，役に立つ（同じ感覚で使えるから）こともあるでしょうし。

Floppy DOSの構造

そろそろ具体的に「Floppy DOS」を見ていきたいと思います。図1を見てください。これは，Floppy DOSとBASICのメモリマップを比べたものです。

はじめに，DOSとはモニタを拡張したようなもの，という説明を行ないました。そのわりにはFloppy DOS本体は13～14Kバイトもあり，モニタの3倍以上にもなります。考え方としては確かにこれで合っているのですが，実際にはディスクBASICから「ステートメント（FORなど）」と「関数」「浮動小数点演算」「グラフィック」などを取ってしまったような構造だと考えてもよいでしょう。ディスクを管理するには，かなり大きなプログラムを要するのです。

ちなみにMZ-1200のH-DOSは，本体1Kバイトの超コンパクトサイズです。したがってユーザーエリアはB•2000のF-DOSくらい残っています。これは，ビルトインコマンド（RENAMEなど，もともと入っている命令と説明したものは，こう呼ばれます）をただのひとつも持たないからです。

Floppy DOSは，表2のようなビルトインコマンドを持っています。ディスクBASICには見られないものも，いくつかあります。

図1-1　MZ-2Z002(カラーディスクBASIC)使用時のメモリマップ

アドレス		内容
0000		モニタ MZ-1Z001M
12A0	BASICインタープリタ	コマンド入力・エディタブロック
199F	BASICインタープリタ	汎用サブルーチンブロック
1DA0	BASICインタープリタ	インタープリタブロック(各種ステートメント)
25FA	BASICインタープリタ	式評価ブロック
2EF0	BASICインタープリタ	浮動小数点演算ブロック
3CE6	BASICインタープリタ	各種入出力ブロック
428A	BASICインタープリタ	ディスク入出ブロック(簡易DOS)
5049	BASICインタープリタ	テープ入出力ブロック
5816	BASICインタープリタ	グラフィックパッケージブロック
65F0	BASICインタープリタ	グラフィックステートメントブロック
6F46		ディスクドライブ制御ブロック
7233		GPIBブロック
7900		RS-232Cブロック
7E00		重要ワークエリア
855C		BASICプログラム(テキスト) 変数エリア 演算用ワークエリア スタックエリア
FF00		システムワークエリア
FFFF		

図1-2　MZ-2Z004(Floppy DOS)使用時のメモリマップ

アドレス		内容
0000		モニタ 2Z-004A
12A0	F-DOS本体	コマンド入力・汎用サブルーチンブロック
2581	F-DOS本体	コマンド処理ブロック
3644		デバイスサポートブロック
4A00		拡張コマンドエリア ユーザーエリア スタックエリア
FE00		システムワークエリア
FFFF		

（モニタ 2Z-004A）MZ-1Z001Mと本質的には同じですが，多少機能の追加がなされています。

図1-3　SB-7020(MZ-80B用F-DOS)使用時のメモリマップ

アドレス		内容
0000		モニタ SB-1521
12A0	F-DOS本体	コマンド入力・汎用サブルーチンブロック
2580	F-DOS本体	コマンド処理ブロック
35DB		デバイスサポートブロック
4900		拡張コマンドエリア ユーザーエリア スタックエリア
FE00		システムワークエリア
FFFF		

図1-4　SP-7010(MZ-1200用F-DOS)使用時のメモリマップ

アドレス		内容
0000		モニタ SP-1002
11D0	F-DOS本体	コマンド入力・汎用サブルーチンブロック
22F4		
22F4	F-DOS本体	コマンド処理ブロック
3146		デバイスサポートブロック
4600		拡張コマンドエリア ユーザーエリア スタックエリア
D000		V-RAM
E000		I/O
F000		フロッピーディスク起動用ROM
FFFF		

（11D0）モニタ用ワークエリアの一部を，F-DOSではプログラム用に使っています。

また，シャープが供給しているトランジェントコマンド（後から拡張する命令と説明したものは，こう呼ばれます）は表3に表わしてあるだけあり，マスターディスクの中に入っています。表3はDIR／Pでプリンタに出力したものです。

ユーザーにとって，表2と表3の命令との差はほとんどないと考えて構いません。なぜなら，CR を押した後にドライブ1を見に行く（本当はロードも行なうのだが，あまりにも速いため「見る」くらいの感じしか受けない）だけだからです。DOSの命令のほとんどはディスクとのやりとりを行なうものですから，1度見に行くくらいの時間はほとんど無視できるのです。

表3にはEDIT（エディタ），ASM（アセンブラ），LINK(リンカ)といった，何となく見たことのある名前があります。これらはMZ-1Z005（システムプログラム）でのそれらをディスク用に少しだけ改造したもの，と考えてよいでしょう。かなり大きなプログラム(？)と考えていたものが，たったひとつのコマンド（命令）になっているのです。

表2　Floppy DOSのビルトインコマンド
（注 MZ-1200では多少内容が異なります）

BOOT	KEY
CHATR	KLIST
CONSOLE	MON
	PAGE
DATE	POKE
DELETE	RENAME
DIR	REW
EXEC	RUN
FAST	TIME
FREE	TYPE
GRAPH	XFER

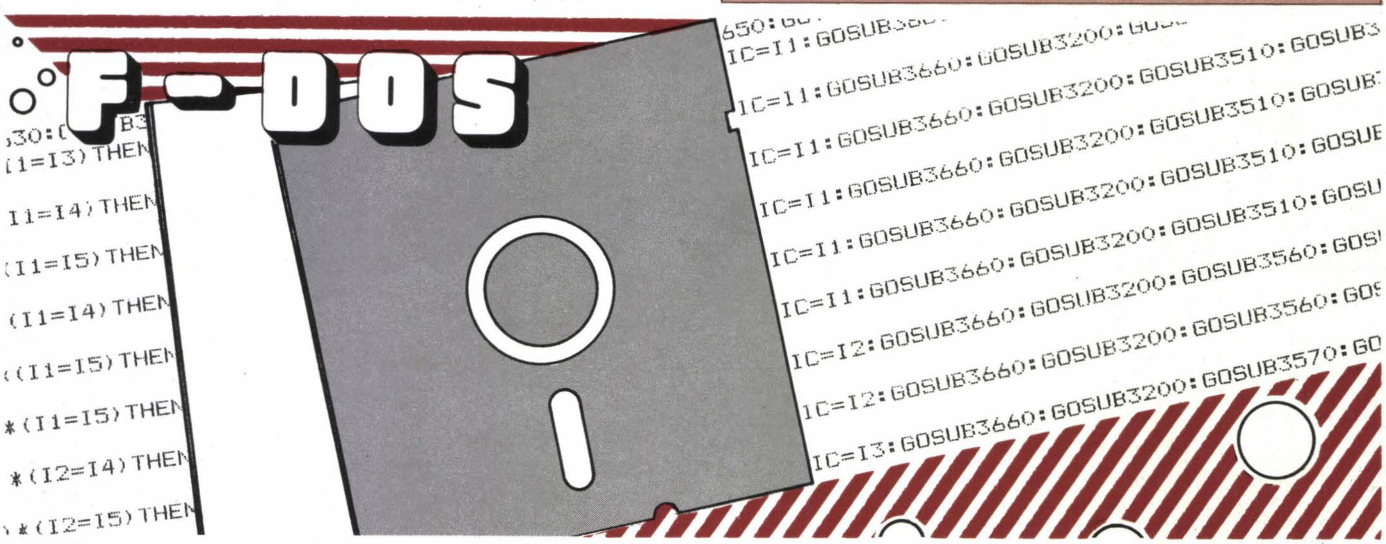
次回予告！

表を見て，どんな命令かな？　って思ったり，さっそく触わってみたいと思った人もいらっしゃるでしょう。次回は，いよいよ各コマンドの説明と，プログラムを作るまでの方法を解説していきたいと思います。ですから，すくなくとも今月の内容はよーく頭に入れておいてください。また，9月号を持っている人は「システムプログラム」の考え方をよく復習してみてくださいね。では来月!!

表3　トランジェントコマンド一欄表

```
  sect AT filename                mm.dd.yy
1>52   OS ASM.SYS                /10.05.82
1>16   OS EDIT.SYS               /10.05.82
1>18   OS LINK.SYS               /10.05.82
1>15   OS MLINK.SYS              /10.05.82
1>24   OS &DEB&.SYS              /10.05.82
1>24   OS PROM.SYS               /10.05.82
1>67   OS BASIC.SYS              /10.05.82
1>64   OS IBASIC.SYS             /10.05.82
1>9    OS FORMAT.SYS             /10.05.82
1>6    OS COPY.SYS               /10.05.82
1>3    OS SIGN.SYS               /10.05.82
1>6    OS VERIFY.SYS             /10.05.82
1>5    OS LIBRARY.SYS            /10.05.82
1>3    OS LOADGR.SYS             /10.05.82
1>3    OS SAVEGR.SYS             /10.05.82
1>4    OS HCOPY.SYS              /10.07.82
1>2    OS LIMIT.SYS              /10.05.82
1>2    OS LOAD.SYS               /10.05.82
1>2    OS ASSIGN.SYS             /10.05.82
1>2    OS STATUS.SYS             /10.05.82
1>24   OS CONVERT.SYS            /10.05.82
1>10   OS PTRP.ASC               /10.05.82
1>3    OS PTRP.OBJ               /10.05.82
1>17   OS SIO.ASC                /10.05.82
1>7    OS SIO.OBJ                /10.05.82
1>25   OS CMT1.ASC               /10.05.82
1>5    OS CMT1.OBJ               /10.05.82
1>11   OS MCR.ASC                /10.05.82
1>3    OS MCR.OBJ                /10.05.82
1>3    OS START-UP.ASC           /10.26.82
1>2    OS LOADAUX.ASC            /10.05.82
1>9    OS MONEQU.ASC             /10.05.82
1>5    OS MONEQU.LIB             /10.05.82
1>13   OS DOSEQU.ASC             /10.05.82
1>8    OS DOSEQU.LIB             /10.05.82
1>6    OS 1Z001MEQU.ASC          /10.05.82
1>4    OS 1Z001MEQU.LIB          /10.05.82
1>107  OS RELO.LIB               /11.12.82
1>71   OS IRELO.LIB              /11.12.82
1>32   OS 2Z004A.RB              /10.26.82
1>2    OS KEYLABEL.ASC           /10.05.82
1>6    OS 8-QUEEN.ASC            /10.05.82
1>21   OS PLOTTER.ASC            /10.12.82
1>23   OS X-YDEMO.ASC            /10.05.82
1>12   OS Centronics.ASC         /10.22.82
1>3    OS Centronics.SYS         /10.22.82
```

（.SYSの付いていないものは別のファイルで，コマンドではありません）



Let's Disk Operating! Part 3

INVITATION TO 'CP/M' (1)
CP/Mってなんだろう

松崎 洋

はじめに

Let's Disk Operating! も早いもので3回目を迎えました。前回まではディスクシステム導入の基本を述べてまいりましたが，ついに今回から何回かにわたりスタンダードバスとしてあまりにも有名な，あのCP/Mをとりあげ，概念から操作上の基本までを順次説明していきます。とにかくこのテの入門記事は実際難しすぎて入門どころでないものが多い（CP/Mそのものが難しいということもいえますが）のですが，読者の皆さんに少しでも役に立つものをということで，初めてCP/Mという言葉を耳にする，あるいは，聞いたことはあるが何のことかわからないという初心者を対象に，Oh! MZ独自の超入門シリーズ風にまとめるつもりです。それから，なにぶん私の勉強不足のため，説明が満足にいきとどかない個所があるかと思いますが，理解できない点，質問などがありましたらぜひお寄せいただきたく存じます。多いお問合せについては，後日誌上を借りて説明させていただきます。

なお，はじめにお断りしておきますが，本稿ではシャープのパソコンMZ-80B/2000用CP/Mを使用して説明しますが，これ以外のハードウェアでも現在CP/Mが走る環境があれば，機種が異なってもCP/Mの働きには何ら変わりありませんので，概要をつかむことは十分可能です。また，今回とりあげるコマンド，仕様などはすべて8ビットタイプ，CP/M-80を対象としています。したがって新製品MZ-5500をはじめ16ビットタイプ用CP/M-86については，（ほとんどCP/M-80と変わりありませんが，細かな相違点について）のちほどまとめて紹介しますので，どうかご了承ください。

CP/Mってなに？

雑誌を眺めていると，よくCP/Mという言葉をみかけます。なんだ，MZと関係ないじゃないか！　と読み飛ばしてしまっていませんか。もし，あなたがそうであれば，たいへん損をしていると思います。

コンピュータは，ハードウェアとともにソフトウェアが必要です。“コンピュータ，ソフトがなければただの箱”などと冗談ぽく表現する人もいますが，実際そのとおりでソフトウェアがなければコンピュータは動きません。MZ-80B/2000，X1そして新製品MZ-5500シリーズなどは電源を入れるとIPLというコンピュータ内部にあるROMに書かれているプログラムが動作して，ユーザーに各種システムプログラム（BASICなど）を起動するための操作を促し，それらをメインメモリにロードします。IPLプログラムは，起動するためのソフトウェアであり，各種システムプログラムが起動すると，もう必要のないものとなってメモリマップ上から外されてしまいますが，仮にこのIPLプログラムがなければBASICを使うことも，また，ゲームを楽しむこともできません。

とかくソフトウェアという無形なものに対して正当に評価されないのは非常に残念なことですが，コンピュータにとってソフトウェアは，このIPLプログラムをはじめとして各種アプリケーションなどハードウェアとともになくてはならない重要なものだといえます。

さて，みなさんはCP/Mをご存知ですか。よく，CP/Mマシンなどと書いてあるものもありますが，CP/Mはソフトウェアですから，正確にはCP/Mというソフトウェアが走るコンピュータをいいます。

では，CP/Mの走る機種とはどんなものがあるのでしょうか。表1にシャープが発売しているパソコンの中で，現在あるいは近い将来発売が予定されているものをまとめました。このように，ディスクシステムが今のところ用意されていないMZ-700シリーズ以外すべての機種で走ります。

CP/Mとは，Control Program for Microcomputersの略称で，シーピーエムと読み，米国デジタルリサーチ社が発売している80系CPU（i8080，Z-80など）用のオペレーティングシステムです。現在8ビット用のOSとしてもっとも普及しているものであり，最近は16ビット機用OS，CP/M-86（8086，8088用）と区別するため，CP/M-80と呼ぶようです。CP/M-86はCP/M-80の機能と特徴をほとんどそのまま受け継いでいます。新製品MZ-5500も16ビット用OSであるCP/M-86を採用，標準装備しています。

それではなぜ，標準OS（DOS）といわれるCP/Mが必要なのでしょうか。

表

機種名	DOSの種類	バージョン	サイズ	供給元
MZ-80K/C	CP/M-80	2.2	48Kバイト	アスキー
MZ-80B/B2	CP/M-80	2.2A	60K	MSA
MZ-2000/2200	CP/M-80	2.2A	60K	MSA
MZ-3500シリーズ	CP/M-80	2.2	60K	シャープ
MZ-5500シリーズ	CP/M-86	1.1	207K㊟	シャープ
CZ-800C/801C/802D	CP/M-80	2.2	60K	シャープ

㊟MZ-5521の場合。

皆さんが，コンピュータを利用するとき，どのような言語を使っていますか。特にシステムプログラムを別途用意しなければ，本体を購入したときに付属してきたBASICしか使えず，かつそのBASIC専用に仕様を合わせて開発された各種アプリケーションプログラムしか利用できません。マイコン用ソフトは必ず○○専用となっていますね。新製品が発表されたからといって，いくらハード的，価格的に魅力であったとしても，発売後果してどのくらいのアプリケーションソフトが開発，市販されるかを考えずに飛びつくと，当初期待していたほどソフトが発売されず，埃をかぶることは必至です。すなわち，完全に孤立してしまいます。メーカーは必ずソフトウェアの互換性をPRしますが，日経パソコン誌（創刊前夏号）で詳しく述べていましたとおり，フルコンパチブルでないとソフトウェア上の互換性はまず期待しないほうが無難でしょう。しかし，世の中には便利なものを作る人がいるもので，このCP/Mを利用すれば，付属のBASICの他にCP/M用に開発された膨大な各種ソフトウェアを利用できるのです。またうれしいことにCP/Mが走っていれば他の機種で何ら変更作業(移植)なしにそのままCP/M用のアプリケーションソフトが走ります。シャープでも，クリーンコンピュータを強調して，BASIC以外にフロッピーDOSやインタプリタPASCALなどを供給していますが，これ以外に，CP/Mを使うと本格的なFORTRAN, COBOL, PL/Iを始めとする高級言語が利用できる，より強力なスーパーマシンに変身します。

CP/Mは，各機種ハードウェアに応じ最小16Kバイトから64Kバイトまでメモリ容量がありますが，CP/M上のアプリケーションソフトウェアによっては，メモリ容量の下限が決められているものも多いので注意が必要でしょう。ただし，MZ-CP/M(80Kシリーズを除く)は，幸いなことにCP/Mとしては最大にランクする60K CP/Mですからあまり関係ないことですが（80Kシリーズは，48K CP/Mです）。

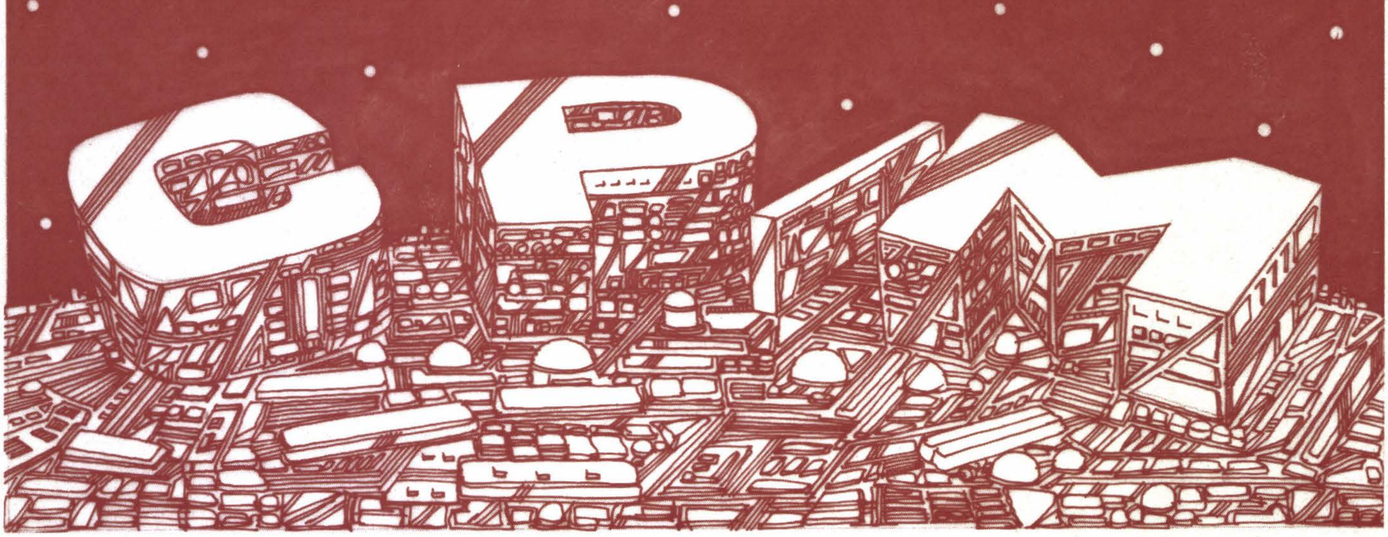
CP/M上で動作するアプリケーションソフトウェアの水準ですが，「マイコン用ソフトだから，低レベルだろう」というのはこの場合に限り違います。たとえば，ホワイトスミス社のCコンパイラはミニコン用OSのUnix Ver. 7のC相当であり，CIS COBOL（MICRO FOCUS社）はANSI 74規格に完全に準拠しているため，大型コンピュータ用のCOBOLに何のひけも取りません。大型コンピュータでプログラムを開発するとランニングコスト（経費）もばかになりませんが，開発，デバッグ作業と手間を要する作業をCP/M（マイコン）上で行ない，大型コンピュータで動作させる，としたらどうでしょう。このように，CP/Mを動かせば，パソコンでオフコンや大型コンピュータと同レベルのソフトウェアが手軽に自宅で動きます。また，CP/Mそのものが一種の開発用ツールとしても使えますので，少し使い込むとアセンブリ言語でプログラミングすることもできます。パソコンではアセンブラに相当するソフトウェアはオプションで用意されているのが普通ですが，これらのソフトはCP/M上のアセンブラに比べ，きわめて低水準なものであり，制約が恐しく多いため，アセンブルやデバッグに結構手間取るものです。

CP/Mに付属するファイルの中で，アセンブルに関する，エディタ，アセンブラ，デバッガの３本のファイルは多機能そして高性能です。エディタ（ファイル名:ED）は大型コンピュータのコンソールと同様な高性能ラインエディタです。アセンブラ（ASM）はインテル形式のアセンブル・ソースをアセンブルし，デバッガ（DDT）はメモリのダンプ，セット，レジスタの変更，逆アセンブラなどの機能をそれぞれ備えています。これらを使うことにより，CP/M上でアセンブリ言語が簡単に扱え，また，必要であれば多機能な開発用ツールも市販されていますので，F-DOSを始めとして，EA-MZ，EDASなどMZ用の安価な低水準「エディタ・アセンブラ」を用いた開発に比べはるかに効率良く行なえるのです。

CP/Mの概要

前項でCP/Mとはどんなことができるかはだいたい理解できたものと思います。しかし，今までBASICという殻に閉じ込もっていた方にとって，ここでは考え方を大きく修正していただかなければなりません。これから説明するCP/Mの概要は，今まで使っていたBASICと次元が異なる別世界ですから，BASICで身についた概念を外してお読みになることをお推めします。

皆さんがCP/Mを始めようとする場合，いくら標準OS，異機種間の互換性がある



といってもCP/Mそのもの（DOS）は，必ず所有する各機種用として発売されているCP/Mを購入しなければなりません。CP/Mそのものは，Digital Research, Inc. の製品であり著作権を保有していますが，全世界でこれだけ多くのパソコンがあると，とても同社だけではフォローしきれないため，日本国内においてはデジタルリサーチジャパンとマイクロソフトウェアアソシエイツ社が代行しています。OEM契約が成立しているMZ-3500，MZ-5500はシャープより提供されています（MZ-3500用CP/M-80 MZ-2Z006はオプション，MZ-5500用CP/M-86は標準装備）が，これ以外の機種においてユーザーの希望によりCP/Mを走らせようとする場合，ユーザーが直接，代理店が発売しているものを求めなければなりません（シングルコピーという）。MZ-80K/C用は，(株)アスキー，MZ-80B/2000用は，(株)マイクロソフトウェアソシエイツが代理店となっています。余談になりますが，デジタルリサーチ（DRI）のライバル会社であるアスキーグループがCP/Mをライセンス契約しているのも実におもしろいことです。

ここでは，MZ-80B（B2）/2000（2200）用MZ-CP/Mを使い，概要を紹介します。その他のCP/Mはアドレス値が異なりますが，内容には支障ありません。

パッケージに含まれるものは，①CP/Mのディスケット，②マニュアル（和文），③契約書2通および返信用封筒，④ファイル（バインダー）となっています。③の契約書とは，今までのソフトと異なり，規定が細かく述べられています。CP/Mはそのソフトウェアを購入するのでなく，使用権を購入すること。このため，使用者契約を供給元と交す必要があるということ。

マイクロソフトウェアアソシエイツ（MSA）版では，契約書の内容に同意ならば契約書の1通を返送（残り1通はユーザーの控え）してほしいとしてありますが，最近の傾向では，同意しない場合はパッケージを開封せず販売店に返せば代金を返却します（MZ-5500用CP/M-86など）と一層きびしくなってきています。契約すれば，各種サポート（バージョンアップ時のアップデイトなど）が受けられますが，同時に契約事項に拘束されます。1ハード1ライセンス（単一の装置に1つのCP/M。仮に3台



のMZ-2000を所有していたとしても3つのCP/Mを用意すること）そして，マニュアルや契約書に定めてある以外のコピー禁止。ユーザーとしては，安く済ませたいところでしょうが，友人と共同購入して共用するといったことは禁じています。

コピー問題が起これば，供給元は裁判も辞さないとしていますので，正規ルートで購入・契約したほうがバージョンアップなどのサポートが受けられ，結果的には有効でしょう。

MZ-CP/Mのメモリマップを図に示しました。さきほども述べましたが，MZ-CP/Mはサイズ60Kバイトを持っているため，パソコン用としては最大クラスを誇っています。そのため，ほとんどすべてのアプリケーションを最大限に利用できます。

図のメモリマップを見てもCP/Mを初めて知る方には良く理解できないと思いますので，各ブロックごと簡単に働きを説明しましょう。各名称は深く考えないほうが無難です。そのうちだんだんとわかってくるものですから。

CP/Mシステムは次に示す4つのモジュールから成っています。

①BIOS（Basic I/O System）

MZのハードウェアとソフトウェアCP/Mとの入出力を行なうI/F（インタフェイス）部分。このBIOSが各々のコンピュータに合わせて独自に作らなければならないハードウェアに依存する唯一のブロック。この部分が各ハードウェアにより異なるため，CP/Mそのものは専用となっています。

②BDOS（Basic Disk Operating System）

入出力の論理的コントロールを行ないます。プログラムやデータは，ファイルとして命名され，ディスク上にディレクトリを持って保存されます。BDOSは新規ファイルの登録をはじめ，ファイルを探し出したり，R/W（リードライト）などをBIOSに命じるわけです。

BIOSとBDOSをFDOS（Functional Disk Operating System）と呼び，CP/MではFDOSがCPUと周辺機器の入出力管理・制御を受け持っています。

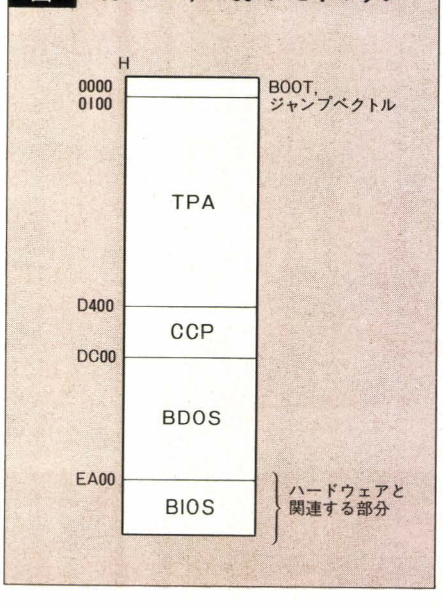
③CCP（Console Command Processor）

ユーザーが入力したキーボードからのデータ内容（コマンド）を解釈する部分。いくつかのプログラムを含み，これらはCP/Mシステム内（メインメモリ）に常駐して，いつでも即実行される状態にあります。このCP/M内蔵のプログラムをビルトインプログラム（コマンド）と呼びます。

④TPA（Transient Program Area）

CP/Mに付属するプログラムがあり，トランジェントプログラムと呼ばれています。これらはディスク上にあり，使用される際呼び出されてこのTPA領域にロードされ，処理されます。CP/Mのコマンドは，この機能を利用し，簡単にいくつでも増やせる

図　60K CP/M-80 メモリマップ



のです。また，各種ソフトウェアがロードされ，実行されるのもTPAであり，ユーザープログラムエリアともいわれます。CP/Mのサイズの違いはこのTPAが大きく取れるか否かの問題だけです。

では，実際に動作のようす（各ブロックの働き）を超スローモーションで考えてみましょう。

ここでは仮に，マイクロソフト社のBASIC，BASIC-80をロードしてみることにします。ドライブAにCP/Mのシステム，そしてMBASIC（BASIC-80）が入ったディスケットが入っているものとします。
キーボードから，

A＞MBASIC [CR]

とタイプインすると，キーボードから入力された内容をチェックするため，BIOSからCCPへ行きます。CCPは内容を解析し，ビルトインコマンドでなければ，ディスクから読み出すためにBDOSへ行きます。BDOSは，ディスクをアクセスし，指定したファイル“MBASIC”を読み出し，TPAにロード，無事ロードが終了すると，TPAにコントロールを移し，MBASICが起動します。MBASICは，自分のプログラムを実行するうえで，CP/MのBDOSをサブルーチンコールしています。

このように，各ブロックは綿密な関係があります。ここでおわかりかと思いますが，CP/M用各種アプリケーションは，ハードウェアに依存するBIOSを使用せず，各CP/M共通であるBDOSをサブルーチンコールとして作成すれば，すべてのCP/M上で何らプログラムを修正することもなく，そのまま使用できることが理解できたものと思います（BIOSをそのまま利用してプログラムを作成するとたとえCP/M用のソフトウェアであっても，その機種専用となってしまう）。

CP/Mは，一種のDOSであり，CP/Mだけでは何の働きもしません。CP/M用として仕様を合わせて開発されたソフトウェアがあってこそ最大限の機能を発揮します。

CP/Mは，標準OSとして地位を確立しているため，仮に膨大な開発費をかけてCP/M用アプリケーションを開発したとしても，世界のコンピュータメーカー500社以上が採用し，100万台以上のユーザーシステムで稼動しているため，十分採算がとれます。良質のソフトウェアが多いわけです。

さいごに

いかがでしょうか。CP/Mの必要性，概要はご理解いただけたでしょうか。

さて，次回ですが，CP/Mに組み込まれているビルトインコマンドをはじめ，CP/Mの基本操作を説明する予定です。なおはじめにも申し上げましたが，わからない点，質問などありましたらぜひお寄せください。誌上でお答えしたいと思います。

本稿を執筆するに際し，下記各文献を参考いたしました。

1）MZ-80B/2200　CP/Mユーザーズマニュアル（Ver. 2.2，2.2A）（株）マイクロソフトアソシエイツ　1982.10
2）標準CP/Mハンドブック　Roday Zaks著，村瀬康治訳，（株）アスキー，1982
3）入門CP/M 村瀬康治著，（株）アスキー，1982
4）Oh！HC Vol.4（株）日本ソフトバンク，1983

今からマシン語を始めてみようというあなた！
もっとマイコンを徹底的に使いこなしてみたいと思っているあなた！
BASICに飽きたあなた！へ送る便利なソフト。

オールマシン語 オリジナルソフト

ユーザーのための便利な機械語ユーティリティー

新発売



機械語動作テストツール

Z80 RELOCATABLE TRACER
リロケータブル　トレーサー

BASICに比べると、機械語プログラムは、それ自身では エラーチェック機能がないため、バグを発見するのが大変です。そこで、効率的に機械語プログラムの開発ができるように、又初心者はZ80機械語の動作を理解できるように、使いやすさを徹底的に追求したZ80リロケータブルトレーサーです。ブレークポインタ方式のデバッガ機能、1命令ずつシミュレートするトレース機能の2つの機能を持ち、自分自身のリロケート機能を利用して、あらゆるアドレス(モニター部、ROM部は除く)でのトレースを可能にしています。

特　徴

① このプログラムは、リロケート機能が付いているので自由なアドレスに配置でき、任意なアドレスで機械語のトレースが可能です。
② Z80シミュレータを内蔵していますので、1命令ずつトレースすることができます。
③ ブレークポインタは32個まで、ループカウンタはそれぞれのアドレスに対して、最大255回までセットすることができます。
④ スクリーンエディット機能をフルに使っていますので、各レジスタの修正、フラグ(ビット単位)の修正が簡単にできます。
⑤ プリンタ出力機能が付いていますので、トレースした結果などをプリンターに打ち出すことができます。
⑥ 逆アセンブル機能が付いています。
⑦ リロケータプログラムが付いているので、各種機械語を自由なアドレスにリロケートすることができます。
⑧ メモリーライト、メモリーダンプ、ブロック転送機能も付いています。

①MZ-700用
②MZ-1200,80K/C用
③MZ-2200,2000,80B用
④パソコンテレビX1用
⑤PC-8001,8001mkⅡ,8801(N-BASICモード)用
⑥PC-8801用

テープ
各￥4,000　〒無料
(マニュアル付)

逆スクロールする
ラベル付
逆アセンブラー

好評発売中



機械語解析ツール

SCROLL DIS-ASSEMBLER
スクロール　ディス　アセンブラー

スクロールディスアセンブラーは、巻物のように上下に自由にスクロールするラベル付Z80逆アセンブラです。

機械語は、普通のダンププログラムでは16進数の表示しかできないので、入力した機械語の確認や、他の機械語プログラムの内容解析が大変です。そこで、このソフトウエアを使用すると、16進数の機械語を分かりやすいニーモニック形式で、又ラベル付きで表示してくれます。また、このプログラムは機械語で作成されているので、高速に解析でき、1つのキー操作で簡単に逆スクロールもできます。

特　徴

① 上下に自由にスクールするので、機械語の解析が楽にできます。
② スクロールスピードを10段階に調整することができます。
③ ラベル付ですので、リストを見やすく出力することができます。
④ リロケート機能が付いているので、自分自身を自由なアドレスに配置することができます。
⑤ オフセット機能が付いているので、他機種(CPUはZ80)のどこのアドレスのプログラムでも、正常に逆アセンブルすることができます。
⑥ プリンター出力機能、アスキーコード出力機能付。
⑦ データエリアを調べて設定することによって、そのエリアはデータとして逆アセンブルリストを出すことができます。

①MZ-700用
②MZ-1200,80K/C用
③MZ-2200,2000,80B用
④パソコンテレビX1用
⑤PC-8001,8001mKⅡ 8801(N-BASICモード)用
⑥PC-8801用

テープ
各￥4,000　〒無料
(マニュアル付)

多機能な機械語
モニター

好評発売中



機械語入力ツール

SELF RELOCATABLE MONITOR
セルフ　リロケータブル　モニター

あなたは、マイコン雑誌から機械語プログラムを打ち込む時、入力方法が分からなかったり、入力ミスをしたり、モニタープログラムを使いにくく思ったことはありませんか？

セルフリロケータブルモニターは各マイコン雑誌から、機械語プログラムを打ち込むための使いやすさを徹底的に追求した機械語モニタープログラムです。

特　徴

① このプログラムは、リロケート機能が付いているので自由なアドレスに配置でき、任意なアドレスへの機械語の入力、LOAD、SAVEができます。
② 機械語を打ち込む時、1行ずつチェックサムを確認しながら入力ができます。
③ 6〜7種類のチェックサム(8バイトごと、16バイトごと、たてよこ(1,2の機種を除く)、128バイトごと、256バイトごと)を出すことができます。
④ キーボードの一部分を16進キー配置に変えて、機械語の入力が素速くできます。
⑤ スクリーンエディタを多くのコマンドで使っているので、修正が簡単にできます。
⑥ プリンター出力機能が付いています。
⑦ 16進、10進、2進の混合加減算機能が付いているので、アドレス計算などができます。
⑧ サーチ機能で機械語の検索ができます。
⑨ リロケータプログラムが付いているので、各種機械語を自由なアドレスにリロケートすることができます。
⑩ テープにセーブされている機械語の各アドレスを調べる機能も付いています。
⑪ 約24種のコマンドをもった多機能機械語モニターです。

(※PC-8001用は、機械語をテープにSAVEする時、ファイルネーム、スタートアドスも記録できるようになっていて、SAVEされた、テープはROMモニターのLコマンドでも、PC-8801モニタROMの"R2: RETURN "コマンド(600ボー)でもLOADできます。)

①MZ-700用
②MZ-1200,80K/C用
③MZ-2200,2000,80B用
④パソコンテレビX1用
⑤PC-8001,8001mKⅡ 8801(N-BASICモード)用

テープ
各￥5,000　〒無料
(マニュアル付)

好評発売中



機械語テープ編集ツール

CASSETTE SAVER
カセット　セイバー

機械語をテープにSAVEしている時、ファイルネーム、アドレスなどが分からなくなったことはありませんか？

カセットセイバーは、カセットテープに記録されている機械語のバックアップを簡単に作成したり、カセットテープの情報(ファイルネーム、各アドレスなど)を調べたり、別のテープに編集しなおしたりできるソフトウエアです。

※編集機能について……本体のRAMをフロッピーディスクのように使用して、機械語、BASIC、データなどを空いているRAMエリアに次々にLOADして、それらのファイルネームを変更したり、順位を入れ換えたり、各アドレスを変更したりして、まとめてあるいは部分的にあるいは何回もテープにSAVEすることができます。

特　徴

BASICプログラムは、リストを表示したり、修正したり、LOAD、SAVEが簡単にできますが、機械語はアドレスが関係しているため、分からなくなると、修正したり、SAVEしたりできません。そこで、このソフトウエアを使用すると、簡単に機械語のファイルネーム、トップアドレス、エンドアドレス、エントリーアドレス、ファイルモードなどを調べたり、LOAD、SAVEしたりすることができます。また、ロードアドレスを変更することもできます。

(ファイルネーム、トップアドレス、エントリーアドレス、ファイルモードの変更。ファイル順位の入れ換え。ファイルのさく除。ファイル一覧表の出力。ファイルのLOAD、SAVE。他。プリンター出力機能付。)

①MZ-700用
②MZ-1200,80K/C用
③MZ-2200,2000,80B用
④パソコンテレビX1用

テープ
各￥3,500　〒無料
(マニュアル付)

★当社の便利な機械語ユーティリティソフトウエアは、次の3つの特徴を持っています。★

1. 特殊なモニターは使わず、標準装備のモニター部を利用しているので、使いやすくなっています。
2. 処理したい機械語とアドレスが重なっていてもよいように、リロケート機能を付けて自分自身のアドレスを自由に変更できるようになっています。(カセット・セイバーはアドレスとは無関係ですので、あえて付けていません。)
3. 約4〜6KBの機械語で作成されているので、処理が高速で、フリーエリアを多く確保することができます。

㈱マイクロラブ

御注文はプログラム名と機種名と電話番号を明記の上、現金書留又は郵便振替でお申し込み下さい。
〒850 長崎市五島町7—17 高島ビル
☎ 0958-27-3725㈹
郵便振替口座 長崎9—2690

MZ-700 (オールマシン語)

SPACE FIGHTER (後)

飯田 賢吾

お待たせしました。前回分がSAVEしてある方は，そのテープをLOADしてくださいね。今回は***M7 FFF**から入力しますが，入力方法についての問いあわせが若干ありました。マニュアルの200ページ以下をよく読んでください。

さて，前回分をLOADしおえたら，*M 7 FFFから打ち始めて，***M930E** で終わりです。

リスト

```
** OH!MZ CHECK SUM SYSTEM V1.0 **

ADR. +0 +1 +2 +3 +4 +5 +6 +7 :SUM.
7FFF 3E 16 CD 12 00 C9 CF 00 :2.CB
8007 13 10 01 03 05 00 06 09 :0.3B
800F 07 08 14 05 12 00 CF 00 :1.09
8017 44 01 03 23 20 20 20 D8 :1.A3
801F 81 87 B3 A4 B4 B5 A6 83 :4.F1
8027 A4 15 06 0F 95 8F 96 BC :3.44
802F 89 8A 98 BC BF 84 BB A2 :5.07
8037 B8 B4 9C BC 87 9A 85 00 :4.6A
-------------------------------------1758
803F 00 92 AE 9C BC BF B0 97 :4.9E
8047 BC 92 BD A3 95 91 86 98 :4.F2
804F 85 84 99 A5 85 8B 92 AB :4.94
8057 B8 BF 92 A2 9E 85 91 BF :5.1E
805F B4 B5 83 A4 84 BC B3 00 :4.83
8067 00 A4 84 97 81 87 B3 A4 :4.1E
806F B4 AA BC A4 A5 85 84 97 :5.03
8077 88 91 BC 98 85 BD 44 94 :4.87
-------------------------------------256D
807F BF 83 AE 87 B2 BF 89 A9 :5.1A
8087 9A 97 92 AB B8 BF 92 A2 :5.19
808F 9E 85 91 BF B4 83 A4 84 :4.D2
8097 BC B3 A4 84 8A 92 BD C4 :5.34
809F 00 C3 00 C2 00 00 00 C1 :2.46
80A7 00 92 AB B8 BF 92 86 BC :4.88
80AF BF 89 A9 BB 97 96 BC BF :5.54
80B7 B0 AA A4 B4 B5 86 BB 84 :5.2C
-------------------------------------2587
80BF 9F 84 8A 92 BD 96 BC BF :5.0D
80C7 B0 AA A4 A0 23 00 00 86 :3.47
80CF B8 9A A4 9A 93 8F 99 A2 :4.ED
80D7 B8 94 BC 21 9A 87 A5 8A :4.79
80DF 92 BD A5 AF AE 87 BC BF :5.53
80E7 8D 9C BC 99 84 B7 A4 84 :4.E1
80EF 97 87 8A 92 BD 00 00 93 :3.8A
80F7 BC 81 8F 94 8B 8F 99 A2 :4.B5
-------------------------------------242D
80FF B8 94 BC 92 AB BC 97 87 :5.1F
8107 A5 AE 93 9C BC BF B0 A0 :5.4D
810F 84 B3 A4 8C B7 A4 97 BC :5.15
8117 92 BD AA B8 85 00 00 99 :3.CF
811F 93 89 A9 9A 97 83 88 93 :4.94
8127 BC 94 BC 8A 84 97 87 8A :4.C2
812F 92 BD CD 86 8D CD 97 8D :5.20
8137 C9 00 22 A1 11 CD 44 00 :2.AE
-------------------------------------2474
813F C9 00 00 00 CD 47 00 C9 :2.A6
8147 00 D5 C5 16 02 0E 00 06 :1.C6
814F 00 10 FE 0D 20 F9 15 20 :2.69
8157 F4 C1 D1 C9 00 00 00 D5 :4.24
815F C5 16 01 0E 00 06 00 10 :1.00
8167 FE 0D 20 F9 15 20 F4 C1 :4.0E
816F D1 C9 00 00 00 F5 C5 E5 :4.39
8177 06 22 21 73 D3 3E 7A 77 :2.BE
-------------------------------------16FE
817F 23 10 FC 00 00 00 06 22 :1.57
8187 21 73 DB 3E 40 77 23 10 :2.97
818F FC E1 C1 F1 C9 00 00 00 :4.58
8197 F5 C5 E5 06 22 3E 60 21 :3.86
819F 73 DB 77 23 10 FC E1 C1 :4.96
81A7 F1 C9 00 00 00 21 10 30 :2.1B
81AF CD 39 81 C9 FF 00 00 C5 :4.14
81B7 0E A0 06 00 10 FE 0D 20 :1.EF
-------------------------------------1880
81BF F9 C1 C9 00 00 00 F5 D5 :4.4D
81C7 CD 3B 00 ED 53 6D 91 D1 :4.17
81CF F1 C9 00 00 00 C5 E5 21 :3.85
81D7 6D 91 7E 87 87 4F 86 77 :3.D6
81DF 79 87 86 3C 77 E1 C1 C9 :4.A4
81E7 00 00 00 D5 11 00 08 19 :1.07
81EF CD D4 81 FE 24 38 1B FE :4.95
81F7 48 38 1B FE 6C 38 1B FE :3.56
-------------------------------------1D55
81FF 90 38 1B FE B4 38 1B FE :3.E6
8207 D8 38 1B 36 10 AF ED 52 :3.5F
820F D1 C9 36 20 18 F7 36 30 :3.65
8217 18 F3 36 40 18 EF 36 50 :3.0E
821F 18 EB 36 60 18 E7 36 70 :3.3E
8227 18 E3 00 00 00 F5 3E FA :3.28
822F 32 7B D1 3E F5 32 7C D1 :4.30
8237 3E 70 32 7B D9 32 7C D9 :3.BB
-------------------------------------1C09
823F F1 C9 00 00 00 F5 C5 D5 :4.49
8247 E5 3E 00 11 06 00 06 11 :1.51
824F 21 53 D0 C5 06 22 77 23 :2.CB
8257 10 FC 19 C1 10 F5 E1 D1 :4.9D
825F C1 F1 C9 00 00 00 FE 11 :3.8A
8267 CC A0 84 FE 12 CC D8 84 :5.28
826F FE 13 CC 52 93 FE 14 CC :4.A0
8277 EA 92 FE 20 CC 70 8A C9 :5.29
-------------------------------------1F7D
827F 00 00 00 AF CD D4 81 FE :3.CF
8287 0F 30 05 DD 21 74 D0 C9 :3.4F
828F FE 1E 30 05 DD 21 9C D0 :3.BB
8297 C9 FE 2D 30 05 DD 21 C4 :3.EB
829F D0 C9 FE 3C 30 05 DD 21 :4.06
82A7 EC D0 C9 FE 4B 30 05 DD :4.E0
82AF 21 14 D1 C9 FE 5A 30 05 :3.5C
82B7 DD 21 3C D1 C9 FE 69 30 :4.6B
-------------------------------------1F71
82BF 05 DD 21 64 D1 C9 FE 78 :4.77
82C7 30 05 DD 21 8C D1 C9 FE :4.57
82CF 87 30 05 DD 21 B4 D1 C9 :4.08
82D7 FE 96 30 05 DD 21 DC D1 :4.74
82DF C9 FE A5 30 05 DD 21 04 :3.A3
82E7 D2 C9 FE B4 30 05 DD 21 :4.80
82EF 2C D2 C9 FE C3 30 05 DD :4.9A
82F7 21 54 D2 C9 FE D2 30 05 :4.15
-------------------------------------221C
82FF DD 21 7C D2 C9 FE E1 30 :5.24
8307 05 DD 21 A4 D2 C9 FE F0 :5.30
830F 30 05 DD 21 CC D2 C9 DD :4.77
8317 21 F4 D2 C9 00 00 00 01 :2.B1
831F 00 08 DD 22 3E 93 DD 56 :3.0B
8327 00 DD 66 FF DD 09 DD 5E :4.63
832F 00 DD 6E FF DD 2A 3E 93 :4.22
8337 C9 AF CD D4 81 FE 80 30 :5.48
-------------------------------------2254
833F 16 DD 36 00 C7 DD 22 3E :3.2D
8347 93 DD 09 DD 36 00 20 DD :3.89
834F 2A 3E 93 CD B6 81 C9 DD :4.A5
8357 36 00 E7 DD 36 FF E8 DD :4.F4
835F 22 3E 93 DD 09 DD 36 00 :2.EC
8367 60 DD 36 FF 60 DD 2A 3E :4.17
836F 93 CD B6 81 C9 E5 DD E5 :6.07
8377 FD E5 F5 21 FF 80 CD 39 :5.7D
```

```
------------------------------------22D6
837F 81 F1 FD E1 DD E1 E1 01 :5.F0
8387 00 08 DD 72 00 DD 56 FF :3.89
838F DD 36 FF C7 DD 22 3E 93 :4.A9
8397 DD 09 DD 73 00 DD 5E FF :4.70
839F DD 36 FF 20 DD 2A 3E 93 :4.0A
83A7 DD 2B CD B6 81 F5 CD 43 :5.11
83AF 81 F1 C9 00 00 00 00 00 :2.3B
83B7 E5 DD E5 FD E5 F5 21 F0 :6.8F
------------------------------------2477
83BF A0 CD 39 81 F1 FD E1 DD :5.D3
83C7 E1 E1 01 00 08 DD 72 00 :3.1A
83CF DD 74 FF DD 56 FE DD 66 :5.C4
83D7 FD DD 36 FE E7 DD 36 FD :6.05
83DF E8 DD 22 3E 93 DD 09 DD :4.7B
83E7 73 00 DD 75 FF DD 5E FE :4.FD
83EF DD 6E FD DD 36 FE 60 DD :5.96
83F7 36 FD 60 DD 2A 3E 93 DD :4.48
------------------------------------280C
83FF 2B DD 2B CD B6 81 F5 CD :4.F9
8407 43 81 F1 C9 00 FF FF 00 :4.7C
840F 00 FF FF F5 C5 E5 D5 CD :6.3F
8417 80 8F D1 E1 C1 F1 C9 00 :5.3C
841F 00 00 00 00 00 00 00 00 :0.00
8427 00 00 00 00 00 00 00 00 :0.00
842F 00 00 00 00 00 00 00 00 :0.00
8437 00 00 00 00 00 00 00 00 :0.00
------------------------------------14F0
843F 00 00 00 00 00 00 00 00 :0.00
8447 00 00 00 00 00 00 00 00 :0.00
844F 00 00 00 00 00 FF FF 00 :1.FE
8457 00 FF FF F5 C5 E5 01 28 :4.C6
845F 00 DD 22 75 84 2A 75 84 :3.1B
8467 ED 42 22 75 84 DD 2A 75 :3.C6
846F 84 E1 C1 F1 C9 00 7E D1 :5.2F
8477 00 F5 C5 E5 01 28 00 DD :3.A5
------------------------------------1679
847F 22 95 84 2A 95 84 09 22 :2.A9
8487 95 84 DD 2A 95 84 E1 C1 :4.DB
848F F1 C9 00 00 00 00 9A D1 :3.25
8497 00 00 FF 00 00 FF FF 00 :2.FD
849F 00 F5 E5 C5 01 22 00 21 :2.E3
84A7 53 D0 3E C7 BE 28 1E 23 :3.4F
84AF 0B 78 B1 20 F5 21 53 D0 :3.8D
84B7 01 22 00 3E E8 BE 28 0D :2.3C
------------------------------------19A1
84BF 23 0B 78 B1 20 F5 C1 E1 :4.0E
84C7 F1 CD BA 91 C9 C1 E1 F1 :6.65
84CF C9 FF FF 00 00 FF FF 00 :4.C5
84D7 00 F5 E5 C5 01 22 00 21 :2.E3
84DF D3 D2 3E C7 BE 28 1E 23 :3.D1
84E7 0B 78 B1 20 F5 21 D3 D2 :4.0F
84EF 01 22 00 3E E8 BE 28 0D :2.3C
84F7 23 0B 78 B1 20 F5 C1 E1 :4.0E
------------------------------------2045
84FF F1 CD 29 92 C9 C1 E1 F1 :5.D5
8507 C9 00 00 FF FF 00 00 FF :3.C6
850F FF 43 43 3B 00 00 00 00 :1.C0
8517 B9 00 00 00 B9 00 00 00 :1.72
851F 00 B9 00 00 B9 00 00 00 :1.72
8527 00 B9 00 00 00 B9 00 00 :1.72
852F 00 00 00 00 00 00 7B 43 :0.BE
8537 43 43 43 3B B9 B9 00 00 :2.76
------------------------------------12E5
853F 00 00 00 00 B9 B9 00 00 :1.72
8547 00 00 00 B9 00 00 00 B9 :1.72
854F 00 00 00 00 B9 00 00 00 :0.B9
8557 00 00 00 00 00 00 7B 43 :0.BE
855F 43 43 43 3B 00 00 00 00 :1.04
8567 00 00 00 B9 00 B9 00 00 :1.72
856F 00 00 00 00 00 00 00 00 :0.00
8577 00 00 00 00 00 B9 00 00 :0.B9
------------------------------------078A
857F B9 00 00 00 00 00 7B 43 :1.77
8587 43 43 43 3B 00 00 00 00 :1.04
858F 00 00 00 B9 00 B9 B9 00 :2.2B
8597 00 00 B9 B9 00 00 00 00 :1.72
859F 00 B9 00 00 00 00 00 00 :0.B9
85A7 00 00 00 B9 00 B9 7B 43 :2.30
85AF 43 43 43 3B B9 00 00 00 :1.BD
85B7 B9 00 00 00 00 00 00 00 :0.B9
------------------------------------0B77
85BF 00 00 00 00 00 00 00 00 :0.00
85C7 B9 00 00 00 00 00 00 00 :0.B9
85CF 00 00 00 00 00 00 7B 43 :0.BE
85D7 43 43 43 3B 00 00 00 00 :1.04
85DF 00 B9 00 B9 00 00 B9 B9 :2.E4
85E7 00 B9 00 B9 00 B9 00 00 :2.2B
85EF 00 00 00 00 B9 00 00 00 :0.B9
85F7 00 00 00 B9 00 00 7B 43 :1.77
------------------------------------09BA
85FF 43 43 43 3B 00 00 B9 B9 :2.76
8607 00 00 B9 00 00 B9 00 00 :1.72
860F 00 00 00 B9 00 00 00 B9 :1.72
8617 00 00 00 00 B9 00 00 B9 :1.72
861F 00 00 00 00 B9 00 7B 43 :1.77
8627 43 43 43 3B 00 00 B9 00 :1.BD
862F 00 00 00 00 00 00 00 B9 :0.B9
8637 B9 00 00 C7 00 00 00 B9 :2.39
------------------------------------0CF2
863F B9 00 00 00 00 00 B9 00 :1.72
8647 00 00 B9 00 00 00 7B 43 :1.77
864F 43 43 43 3B 00 B9 00 00 :1.BD
8657 00 00 00 00 00 00 00 00 :0.00
865F 00 00 00 00 00 00 B9 00 :0.B9
8667 B9 00 00 00 00 00 00 00 :0.B9
866F 00 00 00 00 00 00 7B 43 :0.BE
8677 43 43 43 3B 00 00 B9 00 :1.BD
------------------------------------0893
867F 00 B9 00 00 00 00 00 00 :0.B9
8687 00 00 00 00 00 00 B9 00 :0.B9
868F B9 B9 00 00 00 B9 B9 00 :2.E4
8697 00 00 00 00 B9 00 7B 43 :1.77
869F 43 43 43 3B 00 00 B9 00 :1.BD
86A7 00 00 00 00 00 B9 00 00 :0.B9
86AF 00 00 B9 B9 00 00 B9 00 :2.2B
86B7 00 B9 00 00 00 00 00 B9 :1.72
------------------------------------0BE0
86BF 00 00 00 B9 00 00 7B 43 :1.77
86C7 43 43 43 3B 00 00 B9 00 :1.BD
86CF 00 B9 00 00 00 00 B9 00 :1.72
86D7 00 00 B9 00 00 00 00 00 :0.B9
86DF 00 00 00 B9 B9 00 00 00 :1.72
86E7 00 00 00 B9 B9 00 7B 43 :2.30
86EF 43 43 43 3B 00 00 00 00 :1.04
86F7 B9 00 00 00 B9 00 00 00 :1.72
------------------------------------0B77
86FF 00 B9 00 00 00 00 00 00 :0.B9
8707 00 00 00 00 00 00 00 00 :0.00
870F 00 00 B9 00 B9 00 7B 43 :2.30
8717 43 43 43 3B 00 00 00 00 :1.04
871F 00 00 00 00 00 00 00 00 :0.00
8727 00 00 B9 00 00 B9 00 00 :1.72
872F 00 00 00 00 00 00 00 00 :0.00
8737 00 00 B9 00 B9 B9 7B 43 :2.E9
------------------------------------0848
873F 43 43 43 3B 00 00 00 B9 :1.BD
8747 B9 00 00 00 00 00 B9 00 :1.72
874F 00 00 00 00 00 00 00 00 :0.00
8757 B9 00 B9 B9 00 00 00 B9 :2.E4
875F 00 00 00 00 00 00 7B 43 :0.BE
8767 43 43 43 3B 00 00 00 00 :1.04
876F 00 00 00 00 00 B9 00 00 :0.B9
8777 00 00 00 00 00 00 00 00 :0.00
------------------------------------088E
877F 00 00 00 00 00 00 00 00 :0.00
8787 B9 00 B9 00 00 B9 7B 43 :2.E9
878F 43 43 43 3B B9 00 B9 00 :2.76
8797 B9 00 B9 00 00 00 00 00 :1.72
879F 00 B9 00 00 00 00 00 00 :0.B9
87A7 B9 00 00 00 00 B9 B9 00 :2.2B
87AF 00 B9 00 00 B9 00 7B 43 :2.30
87B7 FF 00 00 FF 80 00 00 FF :3.7D
------------------------------------0F62
87BF FF 57 57 70 70 70 70 70 :3.DD
87C7 20 70 70 70 60 70 70 70 :3.20
87CF 70 10 70 70 70 70 70 70 :3.20
87D7 70 10 70 70 70 50 70 70 :3.00
87DF 70 70 70 70 70 70 70 57 :3.67
87E7 57 57 57 70 20 70 70 70 :2.E5
87EF 70 70 70 70 10 70 70 70 :3.20
87F7 70 70 70 10 70 70 70 40 :2.F0
------------------------------------1979
87FF 70 70 70 70 20 70 70 70 :3.30
```

```
8807 70 70 70 70 70 70 70 57 :3.67
880F 57 57 57 70 70 70 70 70 :3.35
8817 70 70 70 70 70 40 70 70 :3.50
881F 70 70 70 70 70 70 70 70 :3.80
8827 70 70 70 70 70 60 70 70 :3.70
882F 30 70 70 70 70 70 70 57 :3.27
8837 57 57 57 70 70 70 70 70 :3.35
-------------------------------------1A68
883F 70 70 70 20 70 20 60 70 :2.D0
8847 70 70 60 40 70 70 70 70 :3.40
884F 70 30 70 70 70 70 70 70 :3.40
8857 70 70 70 50 70 30 70 57 :3.07
885F 57 57 57 70 50 70 70 70 :3.15
8867 70 70 70 70 70 70 70 70 :3.80
886F 70 70 70 70 70 70 70 70 :3.80
8877 30 70 70 70 70 70 70 70 :3.40
-------------------------------------19AC
887F 70 70 70 70 70 70 70 57 :3.67
8887 57 57 57 70 70 70 70 70 :3.35
888F 70 30 70 40 70 70 40 30 :2.A0
8897 70 60 70 50 70 40 70 70 :3.20
889F 70 70 70 70 10 70 70 70 :3.20
88A7 70 70 70 50 70 70 70 57 :3.47
88AF 57 57 57 70 70 70 40 20 :2.B5
88B7 70 70 50 70 70 60 70 70 :3.50
-------------------------------------18C8
88BF 70 70 70 20 70 70 70 60 :3.20
88C7 70 70 70 70 10 70 70 70 :3.20
88CF 70 70 70 70 10 70 70 57 :3.07
88D7 57 57 57 70 70 70 50 70 :3.15
88DF 70 70 70 70 70 70 70 20 :3.30
88E7 70 70 70 20 70 70 70 10 :2.D0
88EF 70 70 70 70 70 70 10 70 :3.20
88F7 70 70 40 70 70 70 70 57 :3.37
-------------------------------------18B3
88FF 57 57 57 70 70 20 70 70 :2.E5
8907 70 70 70 70 70 70 70 70 :3.80
890F 70 70 70 70 70 70 70 70 :3.80
8917 40 70 70 70 70 70 70 70 :3.50
891F 70 70 70 70 70 70 70 57 :3.67
8927 57 57 57 70 00 00 60 00 :1.D5
892F 00 30 00 00 00 00 00 00 :0.30
8937 00 00 00 00 00 00 20 00 :0.20
-------------------------------------12C1
893F 20 60 00 00 00 60 40 00 :1.20
8947 00 00 00 00 30 00 70 57 :0.F7
894F 57 57 57 70 00 00 10 00 :1.85
8957 00 00 00 00 00 50 00 00 :0.50
895F 00 00 40 20 00 00 50 00 :0.B0
8967 00 60 00 00 00 00 00 20 :0.80
896F 00 00 00 60 00 00 70 57 :1.27
8977 57 57 57 70 70 70 10 70 :2.D5
-------------------------------------0918
897F 70 70 70 70 70 70 10 70 :3.20
8987 70 70 50 70 70 70 70 70 :3.60
898F 70 70 70 20 70 70 70 70 :3.30
8997 70 70 70 10 70 70 70 57 :3.07
899F 57 57 57 70 70 70 70 70 :3.35
89A7 10 70 70 70 40 70 70 70 :2.F0
89AF 70 20 70 70 70 70 70 70 :3.30
89B7 70 70 70 70 70 70 70 70 :3.80
-------------------------------------198C
89BF 70 70 70 70 40 70 70 57 :3.37
89C7 57 57 57 70 70 70 70 70 :3.35
89CF 70 70 70 70 70 70 70 70 :3.80
89D7 70 70 60 70 70 30 70 70 :3.30
89DF 70 70 70 70 70 70 70 70 :3.80
89E7 70 70 20 70 20 60 70 57 :2.B7
89EF 57 57 57 70 70 70 70 60 :3.25
89F7 40 70 70 70 70 70 30 70 :3.10
-------------------------------------1988
89FF 70 70 70 70 70 70 70 70 :3.80
8A07 50 70 30 50 70 70 70 70 :3.00
8A0F 70 70 70 70 70 70 70 57 :3.67
8A17 57 57 57 70 70 70 70 70 :3.35
8A1F 70 70 70 70 70 30 70 70 :3.40
8A27 70 70 70 70 70 70 70 70 :3.80
8A2F 70 70 70 70 70 70 70 70 :3.80
8A37 30 70 40 70 70 40 70 57 :2.C7
-------------------------------------1A23
8A3F 57 57 57 70 30 70 60 70 :2.E5
8A47 50 70 40 70 70 70 70 70 :3.30
```

```
8A4F 70 10 70 70 70 70 70 70 :3.20
8A57 50 70 70 70 70 40 20 70 :2.E0
8A5F 70 50 70 70 60 70 70 57 :3.37
8A67 00 FF FF 00 00 FF FF 00 :3.FC
8A6F 00 DD E5 FD E5 F5 E5 C5 :6.43
8A77 D5 21 50 D0 11 10 85 01 :2.BD
-------------------------------------1C48
8A7F A7 02 ED B0 ED 4B A1 11 :4.30
8A87 ED 43 BA 87 21 50 D8 11 :3.CB
8A8F C0 87 01 A7 02 ED B0 21 :3.AF
8A97 00 01 CD 39 81 3E 76 21 :2.5D
8A9F A3 D1 11 27 00 06 09 77 :2.32
8AA7 19 10 FC 21 A3 D9 06 09 :2.D1
8AAF 3E 10 77 19 10 FC 21 A4 :2.AF
8AB7 D1 06 09 11 29 00 3E 77 :1.CF
-------------------------------------1788
8ABF 77 19 10 FC 21 A4 D9 06 :3.40
8AC7 09 3E 10 77 19 10 FC CD :2.C0
8ACF 60 8C CD AA 93 DD 21 7B :4.6F
8AD7 D1 3E C7 DD BE 00 28 49 :3.E2
8ADF DD BE 01 28 44 AF 00 00 :2.B7
8AE7 00 00 00 00 00 00 00 00 :0.00
8AEF 00 00 00 3E E8 DD BE 00 :2.C1
8AF7 28 75 00 00 00 00 00 2A :0.C7
-------------------------------------1490
8AFF BA 87 CD 39 81 21 10 85 :3.7E
8B07 11 50 D0 01 A7 02 ED B0 :3.78
8B0F 21 C0 87 11 50 D8 01 A7 :3.49
8B17 02 ED B0 D1 C1 E1 F1 FD :6.00
8B1F E1 DD E1 C9 00 00 00 00 :3.68
8B27 00 CD A0 93 AF DD 21 90 :4.3D
8B2F 93 DD 7E 00 DD 86 02 27 :3.7A
8B37 DD 77 00 DD 77 05 DD 23 :3.AD
-------------------------------------1F0B
8B3F DD 7E 00 DD 8E 02 27 DD :3.CC
8B47 77 00 DD 77 03 AF 21 95 :3.33
8B4F 93 ED 67 C6 20 32 21 D0 :3.F0
8B57 AF ED 67 C6 20 32 20 D0 :4.0B
8B5F 21 94 93 AF ED 67 C6 20 :4.31
8B67 32 1F D0 18 44 00 00 CD :2.4A
8B6F A0 93 AF DD 21 90 93 DD :4.E0
8B77 7E 00 DD 86 0A 27 DD 77 :3.66
-------------------------------------1DBB
8B7F 00 DD 77 05 DD 7E 01 DD :3.92
8B87 8E 03 DD 77 01 DD 77 04 :3.3E
8B8F AF 21 95 93 ED 67 C6 20 :4.32
8B97 32 21 D0 AF ED 67 C6 20 :4.0C
8B9F 32 20 D0 21 94 93 AF ED :4.06
8BA7 67 C6 20 32 1F D0 00 00 :2.6E
8BAF 00 00 3E 6B 32 7B D1 32 :2.59
8BB7 7C D1 3E 4D 32 53 D1 3E :3.6C
-------------------------------------1B47
8BBF 42 32 A5 D1 3E 4E 32 55 :2.FD
8BC7 D1 3E 56 32 A2 D1 3E E3 :4.2B
8BCF 32 7A D1 32 7C D1 CD A6 :4.6F
8BD7 0D 3E F9 32 2A D1 32 CE :3.71
8BDF D1 3E F6 32 C9 D1 32 2E :4.31
8BE7 D1 CD 5E 81 CD 43 81 3E :4.4C
8BEF 01 32 9E 93 D1 C1 E1 F1 :4.C8
8BF7 FD E1 DD E1 C9 00 00 3E :4.A3
-------------------------------------20F0
8BFF 00 11 00 00 CD 33 00 C9 :1.DA
8C07 00 00 00 C5 F5 D5 E5 2A :3.9E
8C0F 46 8C CD 3B 00 AF ED 52 :3.C8
8C17 30 14 2A 48 8C 3E 71 77 :2.68
8C1F 2B 22 48 8C 2A 46 8C 01 :2.1E
8C27 09 00 09 22 46 8C 21 32 :1.59
8C2F 01 AF ED 52 38 07 E1 D1 :3.E0
8C37 F1 C1 C3 5C 79 E1 D1 F1 :5.ED
-------------------------------------18EC
8C3F C1 C3 7D 79 00 00 00 09 :2.83
8C47 00 6C DB 00 00 00 F5 AF :2.EB
8C4F 3A 91 93 FE 00 28 05 3E :2.C7
8C57 60 32 B8 81 F1 C9 FF 00 :4.84
8C5F 00 C5 0E 08 06 00 10 FE :1.EF
8C67 0D 20 F9 C1 C9 00 00 00 :2.B0
8C6F 11 06 00 06 11 21 53 D0 :1.72
8C77 C5 D5 06 22 36 C7 CD EA :4.76
-------------------------------------1740
8C7F 81 23 10 F8 D1 19 C1 10 :3.67
8C87 EF C9 00 00 00 2B 46 35 :2.5E
8C8F 2B 47 35 2B 23 44 35 23 :1.91
```

```
8C97 41 35 2B 23 44 37 52 37 :1.C8
8C9F 43 35 44 35 2D 23 41 35 :1.B7
8CA7 2D 46 35 2D 23 41 37 0D :1.7D
8CAF 00 00 00 01 22 00 21 C4 :1.08
8CB7 8C 11 73 D3 ED B0 CD 97 :4.E4
-----------------------------------123E
8CBF 81 C9 00 00 00 00 07 01 :1.52
8CC7 0D 05 00 0F 16 05 12 00 :0.4E
8CCF 00 14 12 19 00 01 07 01 :0.48
8CD7 09 0E 49 68 19 05 13 00 :0.F9
8CDF 0F 12 00 0E 0F 69 00 00 :0.A7
8CE7 00 00 00 00 00 FF 2A E8 :2.11
8CEF 8C ED 5B 90 93 ED 52 38 :4.6E
8CF7 01 C9 2A 90 93 22 E8 8C :3.AD
-----------------------------------0DB4
8CFF 21 90 93 AF ED 67 C6 20 :4.2D
8D07 32 11 D0 AF ED 67 C6 20 :3.FC
8D0F 32 10 D0 21 91 93 AF ED :3.F3
8D17 67 C6 20 32 0F D0 C9 00 :3.27
8D1F 00 00 21 1C DB 22 CE 93 :2.9B
8D27 21 00 00 22 97 93 3E 03 :1.AE
8D2F 32 9C 93 21 09 00 22 46 :1.F3
8D37 8C 21 6C DB 22 48 8C 21 :3.0B
-----------------------------------188A
8D3F 00 00 22 90 93 C9 00 00 :2.0E
8D47 00 21 E8 8C ED 4B E8 8C :4.41
8D4F AF ED 67 C6 20 32 11 D0 :3.FC
8D57 AF ED 67 C6 20 32 10 D0 :3.FB
8D5F 21 E9 8C AF ED 67 C6 20 :4.7F
8D67 32 0F D0 ED 43 E8 8C C9 :4.7E
8D6F FF AA B8 85 99 93 00 00 :4.12
8D77 C7 00 40 25 00 00 E8 E7 :2.FB
-----------------------------------1E50
8D7F 00 40 21 20 00 00 00 3E :0.BF
8D87 20 32 67 DA 3E 60 32 6D :2.D0
8D8F DA 32 6E DA C9 00 00 FF :4.1C
8D97 21 A3 8D 11 D7 D2 01 17 :3.23
8D9F 00 ED B0 C9 0E 05 18 14 :2.A5
8DA7 00 07 0F 49 00 68 08 09 :0.D8
8DAF 14 00 01 0E 19 00 0B 05 :0.4C
8DB7 19 61 69 00 00 00 F5 E5 :2.BD
-----------------------------------1154
8DBF 21 B8 81 AF 3A 91 93 FE :4.65
8DC7 00 28 13 FE 03 38 04 36 :1.AE
8DCF 20 18 0B A7 FE 02 38 04 :2.26
8DD7 36 40 18 02 36 60 E1 F1 :2.F8
8DDF C9 00 00 00 DD 21 0F 8E :2.64
8DE7 DD 36 00 0A 21 D7 D0 06 :2.EB
8DEF 0D 11 1D 00 C5 DD 46 00 :2.23
8DF7 36 7A 23 10 FB 19 DD 7E :3.52
-----------------------------------15F5
8DFF 00 C6 02 DD 77 00 7B D6 :3.6D
8E07 02 5F C1 10 E7 C9 00 00 :2.E2
8E0F 24 00 00 00 3E 77 DD 21 :1.D7
8E17 D1 D8 06 04 DD 77 00 DD :3.E4
8E1F 23 10 F9 DD 77 04 DD 77 :3.D8
8E27 08 DD 77 09 DD 77 0B DD :3.A1
8E2F 77 0C 06 05 DD 77 0E DD :2.CD
8E37 23 10 F9 DD 21 F9 D8 DD :4.D8
-----------------------------------1B28
8E3F 77 00 DD 77 07 DD 77 08 :3.2E
8E47 DD 77 09 DD 77 0C DD 77 :4.11
8E4F 0E 00 00 00 DD 77 10 DD :2.4F
8E57 77 12 DD 21 21 D9 DD 77 :3.D5
8E5F 00 06 03 DD 77 02 DD 23 :2.5F
8E67 10 F9 DD 77 03 DD 77 07 :3.BB
8E6F DD 77 09 00 00 00 DD 77 :2.B1
8E77 0D 06 05 DD 77 0F DD 23 :2.7B
-----------------------------------18A9
8E7F 10 F9 DD 21 49 D9 DD 77 :4.7D
8E87 00 DD 77 03 06 05 DD 77 :2.B6
8E8F 06 DD 23 10 F9 DD 77 07 :3.6A
8E97 DD 77 0B DD 77 0D 06 04 :2.CA
8E9F DD 21 71 D9 06 04 DD 77 :3.A6
8EA7 00 DD 23 10 F9 DD 77 02 :3.5F
8EAF DD 77 06 DD 77 08 DD 77 :4.0A
8EB7 0C 06 05 DD 77 0E DD 23 :2.79
-----------------------------------1AEF
8EBF 10 F9 DD 21 EA D9 DD 77 :5.1E
8EC7 00 DD 77 01 DD 77 02 DD :3.88
8ECF 77 05 DD 77 09 06 05 DD :2.C1
8ED7 77 0B DD 23 10 F9 06 05 :2.96
```

```
8EDF DD 77 0C DD 23 10 F9 DD :4.46
8EE7 21 11 DA DD 77 00 DD 77 :3.B4
8EEF 04 DD 77 06 DD 77 0A DD :3.99
8EF7 77 0C DD 77 12 DD 77 16 :3.53
-----------------------------------1CE3
8EFF DD 21 39 DA DD 77 00 DD :4.42
8F07 77 04 DD 77 07 DD 77 09 :3.33
8F0F 06 05 DD 77 0C DD 23 10 :2.7B
8F17 F9 06 05 DD 77 0D DD 23 :3.65
8F1F 10 F9 DD 21 61 DA DD 77 :4.96
8F27 00 DD 77 04 DD 77 07 DD :3.90
8F2F 77 09 DD 77 0C DD 77 12 :3.46
8F37 DD 21 8A DA DD 77 00 DD :4.93
-----------------------------------1D54
8F3F 77 01 DD 77 02 DD 77 07 :3.29
8F47 06 05 DD 77 0B DD 23 10 :2.7A
8F4F F9 DD 77 0C DD 77 0F DD :4.99
8F57 21 76 DA DD 77 00 C9 FF :4.8D
8F5F FF 00 00 FF FF 00 00 FF :3.FC
8F67 FF 00 00 FF FF 00 00 FF :3.FC
8F6F FF 00 00 FF FF 00 00 FF :3.FC
8F77 FF 00 00 FF FF 00 00 FF :3.FC
-----------------------------------1EB9
8F7F FF 3A 2B D1 21 60 8F 77 :3.BC
8F87 23 3A 2C D1 77 23 3A CB :2.F9
8F8F D1 77 23 3A CC D1 77 21 :3.DA
8F97 77 D1 11 64 8F 01 0A 00 :2.57
8F9F ED B0 21 70 8F 3A 2B D9 :3.FB
8FA7 77 23 3A 2C D9 77 23 3A :2.AD
8FAF CB D9 77 23 3A CC D9 77 :4.94
8FB7 21 77 D9 11 74 8F 01 0A :2.90
-----------------------------------1AB2
8FBF 00 ED B0 3E 3D 32 2B D1 :3.46
8FC7 32 CB D1 3E 71 32 2C D1 :3.AC
8FCF 32 CC D1 3E 6A 21 77 D1 :3.E0
8FD7 77 23 77 23 77 23 23 23 :2.14
8FDF 23 23 77 23 77 23 77 3E :2.2F
8FE7 51 32 7A D1 3E FA 32 7B :3.B3
8FEF D1 3E F5 32 7C D1 3E 57 :4.18
8FF7 32 7D D1 CD D0 79 C9 00 :4.5F
-----------------------------------1B3F
8FFF 00 00 00 00 00 00 00 00 :0.00
9007 00 00 00 00 00 00 00 00 :0.00
900F 00 00 00 00 00 00 00 08 :0.08
9017 09 00 13 03 0F 12 05 4F :0.94
901F 20 20 20 00 00 19 0F 15 :0.9D
9027 12 00 13 03 0F 12 05 4F :0.9D
902F 20 20 20 00 00 00 00 00 :0.60
9037 00 10 20 20 00 20 20 20 :0.B0
-----------------------------------02E6
903F 20 20 20 20 20 20 00 00 :0.C0
9047 40 40 40 40 00 40 40 40 :1.C0
904F 40 40 40 40 40 40 40 00 :1.C0
9057 00 00 D5 F5 11 67 90 CD :3.9F
905F 30 00 F1 D1 C9 00 00 00 :2.BB
9067 46 35 2B 43 37 2B 44 31 :1.C0
906F 2B 43 31 00 23 41 31 00 :1.34
9077 00 2B 43 37 46 35 2B 43 :1.8E
-----------------------------------0F1C
907F 37 2B 44 31 2B 43 31 00 :1.76
9087 23 41 31 00 00 2B 43 37 :1.3A
908F 46 35 2B 45 37 2B 43 35 :1.C5
9097 46 35 2B 45 37 2B 43 35 :1.C5
909F 46 35 2B 45 31 2B 44 31 :1.BC
90A7 2B 45 31 00 00 2B 46 37 :1.49
90AF 0D 00 00 00 00 00 00 00 :0.0D
90B7 00 00 97 BB 87 8F 85 84 :3.71
-----------------------------------0CBD
90BF B3 A4 B5 84 B3 9A 9C BC :5.35
90C7 87 90 A9 61 00 00 00 00 :2.21
90CF 00 00 00 00 00 00 00 00 :0.00
90D7 00 00 00 00 00 00 00 00 :0.00
90DF 00 00 86 BB 81 A6 BF A2 :3.C9
90E7 B8 99 61 00 10 15 13 08 :1.F2
90EF 40 0F 00 00 00 00 00 00 :0.4F
90F7 00 00 00 00 00 00 00 00 :0.00
-----------------------------------0D60
90FF 00 00 00 00 00 A5 99 84 :1.C2
9107 BC 99 92 91 BF 93 61 00 :4.2B
910F 10 15 13 08 40 13 00 00 :0.93
9117 00 00 00 00 00 00 00 00 :0.00
911F 00 00 00 00 00 00 00 00 :0.00
```

```
9127 00 00 00 00 00 A2 8F 85 :1.B6
912F 93 61 00 10 15 13 08 40 :1.74
9137 06 00 00 00 00 00 00 00 :0.06
---------------------------------------09B0
913F 00 00 00 00 00 21 70 8D :1.1E
9147 11 60 D2 01 13 00 ED B0 :2.F4
914F C9 AC BF 91 BC BF A6 99 :5.7F
9157 61 00 10 15 13 08 40 12 :0.F3
915F 00 00 00 00 00 00 00 00 :0.00
9167 00 00 00 00 00 00 07 00 :0.07
916F FF 00 00 B9 B9 00 00 00 :2.71
9177 00 00 00 B9 B9 00 00 00 :1.72
---------------------------------------0E6E
917F 00 00 B9 00 00 00 B9 00 :1.72
9187 00 00 00 B9 00 00 00 00 :0.B9
918F 00 00 00 00 00 00 20 70 :0.90
9197 70 70 70 70 70 70 10 70 :3.20
919F 70 70 70 70 70 10 70 70 :3.20
91A7 70 40 70 70 70 70 20 70 :3.00
91AF 70 70 70 70 70 70 70 70 :3.80
91B7 70 00 00 E5 D5 C5 F5 21 :4.05
---------------------------------------1380
91BF 53 D0 11 72 91 01 22 00 :2.5A
91C7 ED B0 11 53 D0 06 10 C5 :3.AC
91CF 01 22 00 23 23 23 23 23 :0.D2
91D7 23 ED B0 13 13 13 13 13 :2.1F
91DF 13 C1 10 EB 21 53 D8 11 :3.2C
91E7 95 91 01 22 00 ED B0 11 :2.F7
91EF 53 D8 06 10 C5 01 22 00 :2.29
91F7 23 23 23 23 23 23 ED B0 :2.6F
---------------------------------------13B2
91FF 13 13 13 13 13 13 C1 10 :1.43
9207 EB 21 72 91 01 22 00 11 :2.43
920F D3 D2 ED B0 21 95 91 11 :4.9A
9217 D3 DA 01 22 00 ED B0 CD :4.3A
921F A6 0D F1 C1 D1 E1 CD 5A :5.3E
9227 84 C9 E5 D5 C5 F5 21 F4 :5.D6
922F D2 11 94 91 01 22 00 ED :3.18
9237 B8 11 F4 D2 06 10 C5 01 :3.6B
---------------------------------------1DF1
923F 22 00 2B 2B 2B 2B 2B 2B :1.24
9247 ED B8 1B 1B 1B 1B 1B 1B :2.47
924F C1 10 EB 21 F4 DA 11 B7 :4.73
9257 91 01 22 00 ED B8 11 F4 :3.5E
925F DA 06 10 C5 01 22 00 2B :2.03
9267 2B 2B 2B 2B 2B ED B8 1B :2.97
926F 1B 1B 1B 1B 1B C1 10 EB :2.43
9277 21 94 91 01 22 00 11 74 :1.EE
---------------------------------------1407
927F D0 ED B8 21 B7 91 11 74 :4.63
9287 D8 01 22 00 ED B8 CD A6 :4.13
928F 0D F1 C1 D1 E1 CD 78 84 :5.3A
9297 C9 E5 D5 C5 F5 21 54 D0 :5.82
929F 11 53 D0 06 11 C5 1A 01 :2.2B
92A7 21 00 ED B0 12 13 13 13 :2.09
92AF 13 13 13 13 23 23 23 23 :0.D8
92B7 23 23 23 C1 10 E7 00 21 :2.42
---------------------------------------1A80
92BF 54 D8 11 53 D8 06 11 C5 :3.44
92C7 1A 01 21 00 ED B0 12 13 :1.FE
92CF 13 13 13 13 13 13 23 23 :0.B8
92D7 23 23 23 23 23 C1 10 E7 :2.67
92DF CD A6 0D F1 C1 D1 E1 DD :5.C1
92E7 2B C9 00 E5 D5 C5 F5 21 :4.89
92EF F3 D2 11 F4 D2 06 11 C5 :4.78
92F7 1A 01 21 00 ED B8 12 1B :2.0E
---------------------------------------1931
92FF 1B 1B 1B 1B 1B 1B 2B 2B :0.F8
9307 2B 2B 2B 2B 2B C1 00 FF :2.97
930F FF 00 00 FF FF 00 00 FF :3.FC
9317 FF 00 00 FF FF 00 00 FF :3.FC
931F FF 00 00 FF FF 00 00 FF :3.FC
9327 FF 00 00 FF FF 00 00 FF :3.FC
932F FF 00 00 FF FF 00 00 FF :3.FC
9337 FF 00 00 FF FF 00 00 FF :3.FC
---------------------------------------1B77
```

Oh!MZ 質・問・箱

Q X1を買おうと思っています。ところで，X1にはバッテリーのようなものがあり，コンセントを抜いても使えると聞いたのですが，本当なのですか。

愛知県　羽牟　貞幸

A 確かに，X1はバッテリーを持っているのですが，これによってプログラムを走らせたりすることはできません。では，何に使っているのかといいますと，TIME$やDATE$などで使われる，時間や日付けを刻むI·Cのバックアップ電源として使われています。このため，X1では一度時間や日付けをセットすると，停電しても時を刻み続けることになります。

さて，このバッテリーの充電ですが，本体後部のスイッチがONとなっている間は，いつも充電され続けていますので，特に充電に関して気にすることはありません。

Q MZは，POKE文で直接V-RAMに文字を書くことができるそうですが，どのようにすれば書けるのですか。

東京都　板橋　佳也

A POKE文でV-RAMに直接文字を書き込むには，まずそれぞれのコンピュータのV-RAMのセレクト方式を知っておく必要があります。というのは，MZ-80Bや2000はV-RAMがメインメモリの外にあるからです。V-RAMのセレクト方式は機種によって異なるため，それぞれに分けて考えていきましょう。

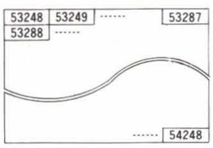


図3

MZ-80K/C/1200

MZ-80K/Cや1200はメモリマップが単純なため（図1-A），特にV-RAMをセレクトする必要はなく，POKEadr, dataとすればV-RAMに文字を書き込むことができます（adrはV-RAMのアドレス，dataはディスプレイコード）。

MZ-700

MZ-700は，電源を入れたときは図1-Aのモードで，S-BASICやHu-BASICを実行しているときには図1-Bの64Kバイトオール RAMモードになっています。このため，80Bや2000と同様にOUT文によってV-RAMをセレクトする必要がありますが，S-BASIC，Hu-BASICともにPOKE文を実行時には，強制的に図1-Aのモードにして V-RAMに書き込みを行なっていますので，特にOUT文などでセレクトする必要はありません。MZ-80K/Cと同じようにPOKEadr, dataとすれば，V-RAMに書き込みます。

MZ-80B/B2

MZ-80Bは，BASICがスタートすると図2-Aのようなメモリマップとなり，V-RAMがメインメモリ上に存在しません。これは0000(H)～FFFF(H)まで64Kバイトをすべてプログラム領域として用いることができるようにと考えられているためです。このため，V-RAMに文字を書き込む場合には，図2-BのようにV-RAMをメインメモリ上にくるようにセレクトしなければなりません。セレクトするには図の下にあるように，OUT文によって切り替えます。OUT @232, 128とすると，以後OUT @232, 0またはPRINT文，CHARACTER$(　)を実行するまで図2のようなモードになり，POKE文で直接V-RAMに文字を書き込むことができます。

このとき注意してもらいたいのは，必ずLIMIT文を実行し，BASIC管理領域を制限してからOUT文を実行しなければなりません（LIMIT$CFFF）。

プログラムは次のようになります。

```
10  LIMIT$CFFF       BASIC領域制限
20  OUT @232, 128      V-RAMセット
30  POKE 53248+X+Y*40, A   書き込み
40  OUT@232, 0      V-RAMリセット
```

40字モード。X，Yは，それぞれX座標Y座標のアドレス。Aは文字データ。

MZ-2000/2200

基本的動作は80Bと同じですが，セレクト方法とセレクト時のメモリマップが図2-Cのように異なりますので，切り替えるときは同図の下に書いてあるように，OUT@232, 192とします。プログラムは，80Bのプログラム20行目をOUT@232, 192とします。

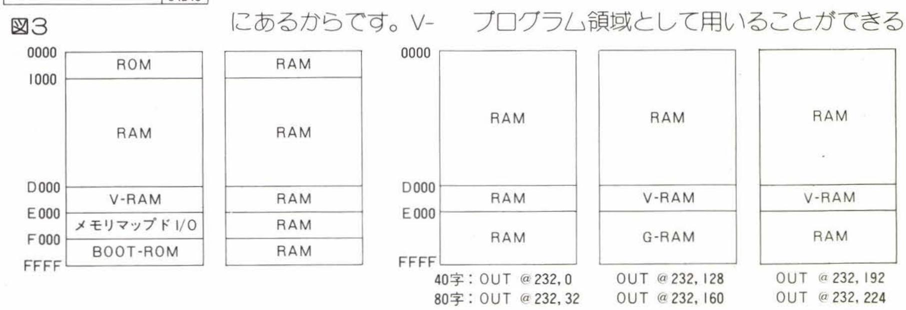


図1－A　1－B　図2－A　2－B　2－C

Q MZ-2200を購入しようと思っていますが，MZ-80Kのモニタ，カセットレコーダとMZ-80 I/Oを接続してMZ-2200に使うことができますか。

岐阜県　古田　裕一

A MZ-80シリーズのディスプレイやカセットを流用したいということですが，残念ながらどちらも使うことはできま

Oh! MZ Q&A

せん。これは，両者のカセットレコーダやディスプレイのコントロール方法および信号が異なっているからです。

MZ-80シリーズのカセットレコーダは，PLAY や EJECT などの動作は手動式で，コンピュータ側からソフトウェアでコントロールできるのは，カセットのモーターのON, OFFとカセットレコーダが動作中であるか否かの検出だけです。これに対してMZ-2200のカセットデッキは，PLAY, EJECTなどすべての動作がソフトウェアでコントロールできるばかりでなく，テープがセットされているか否か，ライトプロテクトノッチがついているか否かの検出もできるようになっています。

これだけ違っていますので，自分で回路を組んでMZ-80K/C用のカセットレコーダを分解して用いるよりもMZ-2200用のカセットデッキ（MZ-1T02）を購入して使用したほうが，信頼性も高く同時に安価であると考えられます。

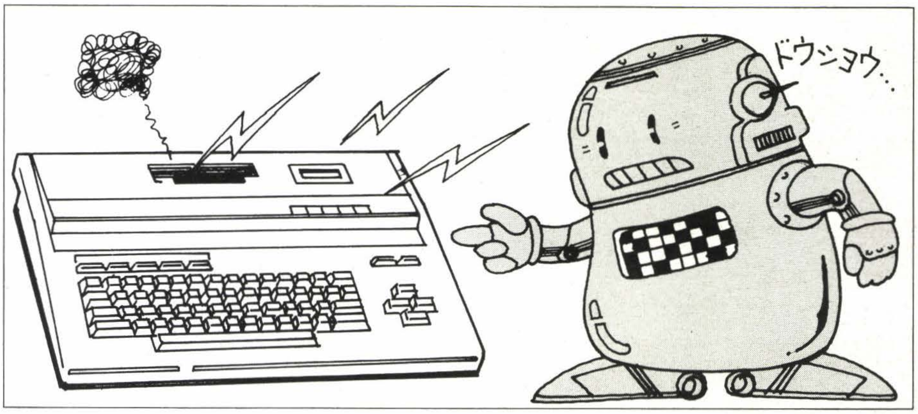
次にディスプレイに関する質問についてですが，MZ-80K/CとMZ-2200のディスプレイの出力信号が同じであれば，流用することができると思います。しかし，これはたいへん危険です。このようなことはさけるようにしてください。

Q 僕は，MZ-700のユーザーです。このごろ電気コードをMZ-700につないで裏の金属（ネジなど）にさわると，電気の流れを感じます。これは故障でしょうか。また，スイッチを入れAB CR と押すと，どのキーを押しても音がするようになります。これは異常でしょうか。

群馬県　荒木　祐一

A 質問からだけでは詳しいことはわからないのですが，あなたのMZは故障しているか，もしくはネジなどが内部電源回路の一部に入ってしまってショートしているのではないかと思います。アースをつけることで直ることもありますが，ひどいようでしたら修理に出すことをお勧めします。

次に，「キー入力時に音がする」という質問ですが，これは異常ではありません。モニタMZ-1Z009Aは，入力した文字列の中に「B」の文字が入っていると，キー入力時に音がなるモードになります。音が鳴って気になるのでしたら，再度「B」を入力してください。音が鳴らなくなります。



Q Oh！MZ 9月号の質問でMZ-2200の周辺機器がMZ-2000ですべて使用できるとのことでしたが，MZ-80Bには使用できるのですか。

神奈川県　本多　富士夫

A いくつかの制限（MZ-80Bは外部にディスプレイがつかないなど）がありますが，プリンタやフロッピーディスクなどは，MZ-80Bにも使用できます。なお，シャープ製品に関する質問は，シャープのサービス部門へ問い合わせれば詳しいことを教えてくれると思います。

MZ-80B専用	MZ-8BK
	MZ-8BG
	MZ-8BGK
MZ-2000専用	MZ-1U01
	MZ-1R01
	MZ-1R02（MZ-2200はすべて実装済）
MZ-2200専用	MZ-1D12
	MZ-1T02

●16bitボードキット（MZ-1M01），16bitボード用漢字ROMボード（MZ-1R08）はMZ-2000，2200専用です。

Q ぼくは，今年の3月にX1を購入し，たくさんのソフトを作ってきましたが，23Kバイトのメモリはちょっと少なすぎます。そこで質問ですが，X1に増設RAMが付けられるのでしょうか。また，将来その可能性はあるのでしょうか。

京都府　大橋　英敏

A さて，X1に増設RAMを付けてユーザーエリアを増やしたいとのご要望ですが，これ以上メモリを増やすことは無理でしょう。それは，X1が増設RAMを付けられるように設計されていないからです。したがって将来発売される予定もないと思われます。

▷▷▷　ひとこと　◁◁◁

MZ-2200/2000（MZ-1Z001M）でコントロールコード（H, C, ⇧, ⇩, ⇦, ⇨など）を直接キーボードから入力するには，

POKE $23, m : POKE $24, n
（m, nはアスキーコード）

とすれば，テンキーの 00 で任意のコントロールコードが入力できます。

質問にお答えします。

日頃疑問に思っていること，どんなことでも結構です。どんどんお便りください。難問，奇問，編集部が総力をあげてお答えいたします。ただし，お寄せいただいているものの中には，マニュアルを読めばすぐに解答が得られるようなものも多々あります。最低限，マニュアルは熟読しておきましょう。質問はなるべく具体的に，機種名，システム構成，必要なら図も入れてこと細かに書いてください。返信用切手同封の質問をよく受けますが，原則として，質問には本誌上でお答えすることになっていますのでご了承ください。なお，質問の内容について直接問い合わせることもありますので，電話番号も明記してくださいね。

宛先：〒102 東京都千代田区四番町2-1
（株）日本ソフトバンク
出版部「Oh! MZ質問箱」係

愛読者 Present

1 Magical Zoo パソコンアニメーター



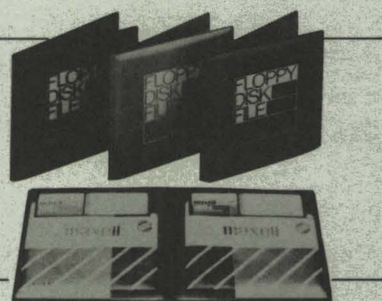
㋐X 1用　カセット版
㋑　〃　　フロッピー版
各3名

自分なりの絵や図形がかんたんに描けるというこの「パソコンアニメーター」。グラフィックカーソルの座標をドット単位で表示し，グラフィックスのモードも表示する。また画面に方眼を表示させせることもできる。画期的なのが，絵の拡大，縮小，転写，保存，呼び出し，プリントアウト機能だ。色もカセット版で40色，ディスク版ではどんな色でも自由に作れる。コマンドもたった20だから，慣れてしまえばLINE文，DATA文で描くのがめんどうくさく思えるだろう。
要V-RAM，FD版9,800円　カセット版5,800円

2 コクヨ フロッピーディスクファイル

各10名

5インチフロッピーディスク2枚を収容できるフロッピーディスクファイル<ED-F11>。新発売を記念して10名の方にプレゼント。
表紙にウレタンを入れてあるので，外部からの衝撃にも安心して持ち運びできる。サイズはタテ155×ヨコ158×背15ミリ。500円



3 マイクロラブ



A. Z80リロケータブルトレーサー
B. スクロールディスアセンブラ
C. セルフリロケータブルモニター

㋐MZ-700　㋑MZ-80K/C/1200
㋒MZ-2000/2200/80B　㋓X 1
各3名

これからマシン語を勉強したいというあなたに贈る，便利な機械語ユーティリティ。Aはリロケート機能付きの機械語動作テストツール。Bは上下に自由にスクロールできる機械語解析ツール。Cはリロケート機能付き機械語入力ツール。いずれも約4～6KBの機械語で作られているため，処理速度が速く，フリーエリアも多く確保できるという便利物だ。
A，B．各4,000円　C 5,000円 詳細なマニュアル付き。

4 Carry lab. HIROTON WARS

㋐MZ-2000/2200用
㋑X 1用
各3名

今，人気のスーパーリアルタイムの3Dタンクゲームが，「ヒロトンウォーズ」だ。
敵ミサイル，地雷を避けて，敵戦車を破壊するという，よくありそうな内容だが，画面が3D処理で，なかなかの迫力。さて，君はいくつのキャラクタを見ることができるか？チャレンジングステージが君を待っている。
MZ-2000/2200用 3,500円 X 1用 3,000円
いずれもカセット版



プレゼントの応募方法

とじ込みのアンケートはがきの該当項目をすべてご記入のうえ，希望のプレゼント番号をはがき右下のスペースにひとつ記入(3－A－アなど)してお申し込みください。締め切りは11月15日の到着分までとします。当選発表は1月号で行ないます。

9月号プレゼント当選者発表

①㋐（茨城県）篠塚幸男（大分県）阿南厚己（埼玉県）佐藤卓士（大阪府）新谷順二（熊本県）福浦隆生（埼玉県）関口光英（和歌山県）西門秀人（三重県）中村敏文（神奈川県）米山善一（北海道）宮川満様ほか10名　㋑（長野県）杉戸雅敏（千葉県）石川信幸（北海道）久保昌弘（青森県）織田仁志（神奈川県）佐藤心美（山口県）大畠金槌（京都府）玉田コスモ（栃木県）菅野哲也（大阪府）竹林清（岐阜県）二村洋一様ほか10名　㋒（広島県）久田将司（千葉県）小仁所利男（石川県）綿谷辰夫（大阪府）坂本充也（高知県）酒井達則（神奈川県）島崎勇（東京都）金庸豪（富山県）橋本均（大阪府）金重浩道（石川県）野村学様ほか10名　②㋐（埼玉県）熊倉広（山形県）進藤拓（千葉県）川崎雄太（埼玉県）岡田典倫（千葉県）川村恵介　㋑（愛知県）畑中秀樹（群馬県）小堀勝利（静岡県）石岡宏康（大阪府）川口雅弘（兵庫県）木村浩也　㋒（静岡県）斉藤直秀（岡山県）赤木有（神奈川県）川名篤（北海道）長岡朋浩（広島県）尾崎弘樹　㋓（石川県）北出一彦（富山県）宮村孝廉（山口県）田島信彦　③（福井県）山口賢剛（鹿児島県）田中美年（島根県）吉岡邦彦（埼玉県）舟橋高保（東京都）貝瀬宏　④㋐（岩手県）田波俊一（福岡県）橋口三郎（宮崎県）小笠原純一　㋑（香川県）造田英俊（群馬県）鈴木秀一（宮城県）中村昌央
以上の方々が当選されました。おめでとうございます。

マイコンピュータテレビC1

――ホームコンピュータ&テレビ――



コントローラを利用して絵が描ける。字が書ける。算数もできるし，ゲームもできる。マイコンを組みこんだユニークなテレビがシャープから新発売されました。

(1) ジュニアグラフイック機構とコントローラによって画面を上下20,左右20のマスに分け，その1つひとつに好きな色や好きな形のパターンを入れて絵がかけます。使えるパターンは60種類，12色。背景色は5色です。完成した絵はスイッチを切っても，もう一度再現できるほか，別売のカセットインターフェースによってカセットテープに記録することも可能です。

(2) テレビノート機構は，1行28字までの文字を11行。記号はカタカナ，アルファベット，時間や曜日を表わす漢字，数字など134種類。メッセージを一時的に保存したりカセットテープに記録したりできます。

(3) 算数ゲーム機構は，ゲームしながら四則計算を練習します。ドンキーコングジュニアを上下左右に操作して，正解を出しましょう。1回10問。4種類の計算とケタ数を選べます。

(4) もちろんゲーム機構でテレビゲームが楽しめます。コントローラは2個が付属していますから，2人でゲームを楽しめますし，1個はマイクつきなので歌をうたったり音楽レッスンに使えます。ゲームカセットは任天堂株式会社のファミリーコンピュータ用のカセットが利用できます。

カラーテレビジョンの仕様は下記のとおりです。色は，ゴールド，レッド，ホワイトの3種。価格は19型の19C−C1F・Wで145,000円，14型の14C−C1F・W・Rで93,000円です。

形　名	受信周波数	ブラウン管	使用ブラウン管の大きさ	スピーカー	音声出力	消費電力	寸　法	重　量
19C-C1F・W (卓上型)	1～62チャンネル	19型90度偏向 シャープSOブラウン管	幅36.6cm・高さ27.4cm	10cm丸型1個	1.3W	101W (キャンバスホールド時6.5W)	幅47.6cm 奥行44.5cm 高さ45.1cm	19.2kg
14C-C1F・W・R (卓上型)	1～62チャンネル	14型90度偏向 ハイコントラストブラウン管	幅28.1cm・高さ21.1cm	10cm丸型1個	1.3W	71W (キャンバスホールド時6.5W)	幅38.1cm 奥行38.0cm 高さ37.7cm	11.8kg

●MZ試写室●絵そらごとではすまされない，今，全米で話題の映画

ウォー・ゲーム WARGAMES

MGM／UA映画
CIC配給



アメリカの高校生が自分のパソコンを学校や病院などのコンピュータとアクセスして，成績，カルテを書き換えてしまった…というニュースを耳にした読者も多いだろう。「え？　そんなことが…」と思う方もいるだろうが，当の高校生に言わせると，「電話回線で継ぐことはそんな大したことじゃないし，パスワードにしたって，いかにもありそうなものを使ってる場合が多いからかんたんなものさ」との弁。

こういったコンピュータを利用した事件が、今アメリカで何件か起こり，問題になっている。折りも折り，これらの事件と似た内容の映画が封切られていたのだ。

恐い話だが、限りない可能性が

その話題の映画が「ウォー・ゲーム」。

デビッドはシアトル郊外に住む17才の高校生。勉強は並だが，コンピュータにはめっぽう強い。パソコンゲームにはあらかた飽きてしまって，最近では，学校のコンピュータと自分のパソコンとをつないで，成績データを書き換えたりする悪ガキぶりだった。

ある日，雑誌の広告でみつけた新しいパソコンゲームの内容を探ろうと，電話回線を使ってゲーム製造会社のコンピュータを呼び出した。そのゲームリストの中には，チェスやチェッカーのほか，生化学戦争とか〝世界全面核戦争〟など気を引きそうなゲームがいくつか入っていた。

さんざん苦労して探し出したパスワード〝ジョシュア〟で「世界全面核戦争」のゲームをアクセスし，デビッドは夢中になって遊び始めた。コンピュータがアメリカ軍，デビッドがソ連軍を引き受け，デビッドはミサイルをラスベガスやシアトルめがけて発射したのだ。

SHARPから新型ポケコン

PC-1255を発売

シャープのポケコンシリーズに新しいタイプが加わりました。携帯に便利な手帳タイプですが，RAM容量10.2KバイトのPC-1255です。もちろん，姉妹機のPC-1250/1251とコンパチで，豊富なソフトウェアがそのまま使えます。価格は35,800円です。

特長

(1) ROM24Kバイト，RAM10.2Kバイト，メモリ保護機能つきです。データ容量の多いプログラムやステップの多い高度なアプリケーションへの対応が，手帳サイズで実現できます。

(2)PC-1250/1251のBASICを採用して，蓄積された豊富なプログラムが活用できます。

(3) 表示部は24桁液晶ドットを採用。

(4) 使用頻度の高い数字キー，ファンクションキーを大きくして，操作性を高めました。

(5) 18個のアルファベットキーに，キー操作を記憶させるリザーバルキー，プログラムの頭出しができる定義づけキーを採用しています。



Z8000用オペレーティングシステム UCSD p-Systemを発売

シャープでは，JBA（日本ビジネスオートメーション）社との技術提携によって，Z8000 16ビットコンピュータ用のUCSD p-Systemを発売することになりました。

UCSD p-Systemは，CP/M（デジタルリサーチ社，MS-DOS（マイクロソフト社）の並ぶオペレーティングシステムで，1974年にカリフォルニア大学サン・ディエゴ校（UCSD）で開発され，UCSD Pascalをはじめ，FORTRAN, BASICなどの言語が使用でき，エディタ，ファイラなどを含んでいることから，初心者にも経験者にも使いやすいトータルシステムです。

このシステムを利用することによって，Z8000ファミリーはより利用しやすくなるでしょう。



ところが，これが世界を揺るがすトラブルの始まりだったとは！　なんとデビッドのパソコンは，，NORAD(北米大陸防空軍)の作戦コンピュータWOPRに接触してしまったのだ。

同じころ，NORAD 戦略作戦室はハチの巣をつついたような大騒ぎとなっていた。作戦室の大スクリーンには，ソ連がミサイルを発射したとの表示とともに緊急警報が鳴りひびいてきたからだ。

デビッドがゲームだと思って楽しんでいたのは，実はアメリカ国防省の核戦略コンピュータのプログラムだったのだ。TVニュースでこの事実を知った彼は，青くなってゲームを中止しようとした。しかし，〝ジョシュア〟は勝手に彼にゲームの続きをやろうと要求してくる。そう，このプログラムは最後まで続行するようプログラムされているのだった。

国防省の大騒ぎは，単なるコンピュータのゲームなのか，はたまた……。





コンピュータ万能時代への警鐘

この映画，アメリカで今年6月中旬に封切られ，あまりのリアリティ，恐怖感に轟轟たる反響を呼んだ。冒頭で書いたニュースも，この映画の影響と言えなくもない。映画の中の絵そらごとが現実として起こりうる，いや実際に起こっているというリアリティぶりに，背すじに寒いものが走った。

映画の冒頭でアメリカ国防省のコンピュータ管理責任者と軍の指揮官が激論を交わしていた。核ミサイル発射のボタンを押すのは人間が良いか，コンピュータがよいか…。コンピュータはさまざまなデータをもとに的確な判断でボタンを押す。しかし，人間には心があると。

結局，結論が出ないまま，このトラブルに巻き込まれるわけだが，結論は火を見るより明らかだと思うのは私だけだろうか。やっぱりコンピュータを使う人間でいたいからね。

監督は，本編とともに最新テクノロジーの脅威を描いた話題作「ブルー・サンダー」でニュー・エンタテイメントの旗手と目されるジョン・バダム。CICの配給で，日本での封切りは12月初旬の予定だ。くれぐれも心して見られよ。(K)





miniHuBASICコンパイラ

整数型BASIC⇒マシン語変換プログラム

ハドソンから,新しいコンパイラ型のBASIC(MZ-700／X１用)が発売されました。定価が6,000円と，少々高めのようですが,そのあたりのことを含め，使ってみた感じをレポートしたいと思います。

HuBASICとの違い

この言語ソフトは，HuBASICコンパイラ(MZ-80K/C/1200用)を小さくしたもの,という意味ではなく，新しいミニHuBASICという言語のコンパイラです。

ミニHuBASICと付属のHuBASICとの違いは,だいたい次のようになるようです。

●HuBASICは1.5のような小数や10^{10}のような大きな値(単精度実数，倍精度実数)が扱えましたが，ミニHuBASICでは−32768〜＋32768の整数しか使えません。また，SIN，TAN，ATN，SQR，EXP，LOGのような数値計算用関数もなくなりました。

●HuBASICではA＄＝"Oh!MZ" のように文字変数を使うことができましたが，ミニHuBASICでは文字変数はありません。ただし，PRINT文中で" " で囲めば，文字を表示させることはできます。そのため，RIGHT＄，LEFT＄,MID＄,CHR＄，STR＄，ASC，VAL，LENなどのすべての文字関数がなくなっています。

●BASICとしては極めて初歩的で利用頻度の高い，FOR〜NEXT文，INPUT文，READ〜DATA文，LPRINT文が使えません。ですからこれらを使いたいときは，もう少し難かしい命令をたくさん組み合わせて実現させるしかないようです。

●X１のHuBASICにはPSET，LINE，CIRCLE，PAINTなど，数多くのグラフィック命令がありましたが，X１用ミニHuBASICにはPATTERN文しかありません。したがって点を打ったり，線・円を描いたり，ぬりつぶしたりはできません。

●X１特有のカセット・テレビ・多彩なキャラクタ制御（CSIZEなど)，また，MZ-700のプロッタプリンタ命令，タイマー制御(TIME＄など)，音楽(PLAY文，MUSIC文，TEMPO文)などはできません。ただ，X１ではDEFCHR＄とSOUNDだけは使えます。

●HuBASICで作成したテープは，ミニHuBASICでLOADしたりはできません。この逆も否です。

●変数名は，HuBASICでは255文字までの任意の長さでしたが，ミニHuBASICは１文字のみです。ですから，変数は26個までしか使えません。配列も使えますが，メモリの割り振りを自分で決めなければなりません。ということは，ある程度のマシン語の知識がないと，A(10)などは使えないということです。

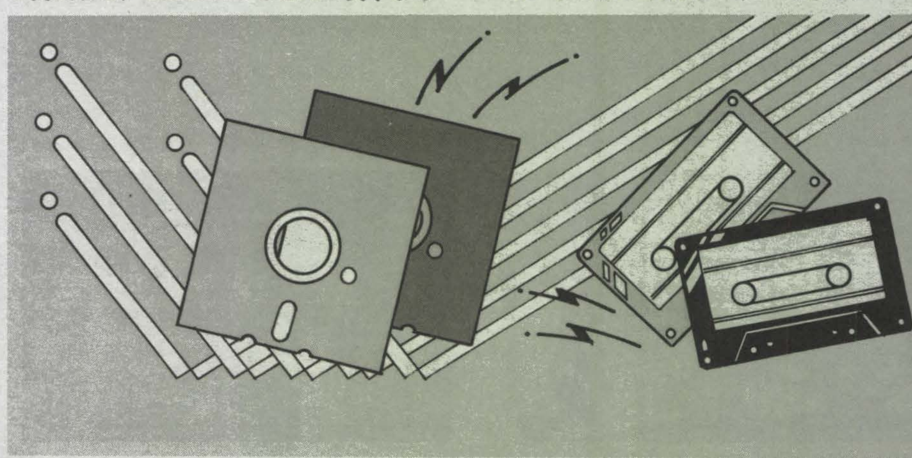
●関数でありながら，引数（変数に限る）の値を＋1，−1できるINC，DECの両関数が追加されています。

●スピードが速くなります。コンパイラは，一度プログラムをマシン語に直し，そのあとで一挙に実行をします。このスピードアップこそが，このソフトの最大のメリットです。

というわけで，かなり異質なBASIC――というよりWICS*のような完全に別の新言語である，と言ったほうが良いかもしれません。

X１で，HuBASICとミニHuBASICコンパイラとのスピードを比較してみるために，一般的なプログラムを組んでみました。

TIME＝０:FORA％＝0TO32766:NEXT:PRINT TIME　(HuBASIC)



A＝0 :REPEAT :UNTILINCA＝32766
(ミニHuBASIC。計時機能がないため,ストップウォッチで測定)

見ておわかりのように，ほとんど別の表現になっています。

ちなみに，このテスト実行結果は，

HuBASIC……10秒

ミニHuBASIC……1.5秒

で，約6.7倍のスピードアップになりました。

ミニHuBASICは「コンパイラ」ですから，前に言ったように，一度すべてをマシン語に直してから実行に移ります。したがって，元のプログラムにバグがあると，生成されるマシン語にもバグが受け継がれ，実行に移した時点で暴走したりします。インタプリタでないため，エラーを出して警告してくれないのです。

これではデバッグがほとんどできないということで，このソフトにはインタプリタも付いてきます。使える命令はコンパイラと同じで，スピードは遅く（テストプログラムは15秒もかかった），エラーチェック機能が付いているという仕様です。

また，IPLから直接ロードして実行できるようにするユーティリティも付いてきます。

＊キャリーラボ製の言語で，ミニHuBASICとかなり似た内容を持っている。

評　価

このミニHuBASICコンパイラ，数々の制限を加えたことで，かなりのスピードを実現してくれます。マシン語を知らない人でもスピードの速いゲームが作れ，しかも作る時間も短かくて済むという点では，6,000円という価格は，決して高くないかもしれません。ただ，せっかくなら，従来のHuBASICと完全にコンパチブルなコンパイラにしてくれれば良かったのですが……。そうすれば，「ミニHuBASIC」独自の決まり文句，プログラム作りのコツなどを改めて発見・勉強し直さなくても済むわけですからね。

整数専用で文字列・グラフィックがダメとあっては，応用する範囲も限られてしまい，せいぜいキャラクタを使ったリアルタイムゲームくらいにしか利用できないでしょう。しかし，それに限って考えれば，まあ使いモノになりそうです。

実用性を考えるならば，やはりホンモノの「HuBASICコンパイラ」に期待したいところです。　（渡辺　健）

図1　HuBASICとミニHuBASICコンパイラ，インタプリタの関係

HuBASIC
整数
実数（変数，関数）
文字（変数，関数）
INPUT, READ～DATA,
FOR～NEXT, LPRINT など
グラフィック制御
カセット・テレビ制御
（MZ-700はプロッタプリンタ制御）
PLAY, MUSIC, TEMPO
変数は任意個で，任意長名
スピードは遅い
キーボードからプログラムを入力できる
RUNで実行する
エラーを検出できる

↔テープフォーマットが異なる（互換性なし）

ミニHuBASIC
整数
グラフィックはPATTERNのみ
変数はA～Zの26個
スピードは遅い
キーボードからプログラムを入力できる
RUNで実行する
エラーを検出できる

HuBASICコンパイラ
発表されていない
（MZ-80K/C/1200にはある）

ミニHuBASICコンパイラ
整数
グラフィックはPATTERNのみ
変数はA～Zの26個のみ
スピードは速い
キーボードからプログラムを入力できる
COMPILEでマシン語に直し，EXECで実行する
エラーを検出できない

テープフォーマットが同一（互換性あり）

表1　ミニHuBASICで使える命令など

LOAD	LIMIT	DEFCHR$(X1のみ)	DEC
LOADM	GOTO	PATTERN(X1のみ)	NOT
SAVE	GOSUB	END	INP
SAVEM	RETURN	SOUND(X1のみ)	ABS
LIST	PRINT(？でもよい)	REPEAT～UNTIL	STICK
LLIST	LOCATE	COLOR	STRING
DUMP（メモリダンプなど）	POKE	SEARCH	+，−，＊，/
RUN（インタプリタのみ。	IF～THEN(ELSEはない)	PEEK	AND
コンパイラでは，	CLS	RND	OR
COMPILE　$4000⏎	PAUSE	KEY(INKEY$に相当)	XOR
EXEC⏎のようにする）	CALL	GET(INKEY$(1)に相当)	>, <, =, >=, <=, <>
NEW	OUT	MOD(使い方が異なる)	
BOOT	WIDTH(X1のみ)	INC	

X1のグラフィック機能を生かしたCGツール

最近はCG（コンピュータ・グラフィックス）がなにかと話題になっています。

グラフィック関係の単行本もいくつか見かけるようですし，東京ではコンピュータグラフィック展なども催されています。

CGによる映画「ゴルゴ13」なども出ていますが，これらの写真のようなあざやかなグラフィックは，大型コンピュータの高速処理と膨大な記憶量のメモリを使用してはじめて実現できたわけです。

マイクロコンピュータといえどもコンピュータには違いないのです。大型コンピュータには及びませんが，マイコンでもなかなかの絵が描けるはずです。

特にX1は640×200ドットのすべてに8色の色を指定できるという，強力な機能を持っています。

今回紹介するソフトは，X1のグラフィックを生かして，ディスプレイ画面をキャンバスにCGを描くためのソフトです。

ただ単に絵を描くだけでなく，テープやディスクに記録して残しておけますし，ゲームのデモンストレーションやアドベンチャーゲームの絵としても応用できます。

さて，このようなソフトを選ぶにあたって気になる点は，1)どのような機能を持ち，2)どれだけ使いやすいか，の2点でしょう。

つまり，どれだけ手軽に思いどおりの絵が描けるかが問題なのです。

まず，機能について説明を加えながら個条書きにしてみます。

①ラインを描く

プログラムをスタートさせると，画面上にふたつの点滅する光点が出ます。一方は□，他方は＋の形をしています。テンキーやジョイスティックを使って我々が動かせるのは＋の方で，こちらをメインに考えてください。

Lを押すことによって□と＋の間にラインが引かれます。

このとき，引かれた線が気に入らないものは取り消すこともできます。

あるコマンドを使うたびに確認のメッセージが出てきます。この点は親切な設計といえるでしょう。

ただし，線を引くか取り消すかの判定のときNキー以外はYESと判断されてしまいます。つまり，Lキーを長く押していると勝手に線を引いていってしまいます。

ラインを描く方法にはもうひとつあり，＋の動いた軌跡が線として引かれるもので，曲線を引くのに有効でしょう。

ラインの色は8色指定可能です。

②ペンの色を変える

③＋を10倍速く移動させる

④＋を指定した点に移動させる

これは＋の座標を数値で指定することができます。原画の座標を読んで忠実に描くときに使えるでしょう。

⑤　BOXを描く

⑥　半円を描く

□と＋を直径とした半円を描くもので上半円と下半円を指定できます。

⑦　円を描く

これは中心のX・Y座標，半径，円を描き始める角度と，終わりの角度を指定していろいろな円を描くことができます。

⑧　色を塗る

使える色はすでに40色用意されており，中間色なども豊富です。さらに60色を自分で作ることができます。

色を塗る場合，完全に線で囲まれていないと色が外へ出てしまい，今まで描いてきた絵がだいなしになってしまいます。

色がもれたときの対策も用意されていますが，もう少し完璧な配慮をしてもらいたかったところです。

⑨　色を作る

⑩　絵を記憶する

この命令で描いた絵は，好きなところに好きな大きさでいくつでも描くことができます。

⑪　フレームをつける

⑫　文字を書く

ノーマル・横長・縦長の文字をグラフィック上にかさねて表示する。

⑬　長方形単位で色を変更する

これはBOXとペイントをあわせたもので，色もれの修正にも便利に使えます。

⑭　画面のセーブ・ロード

⑮　画面のハードコピーをとる

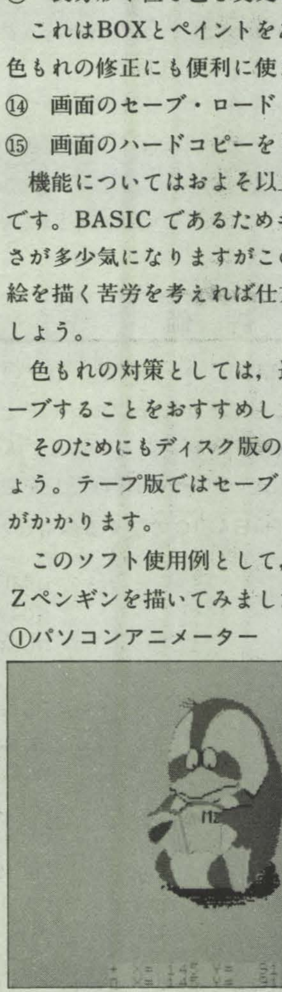
機能についてはおよそ以上のようなものです。BASICであるためキーの反応の遅さが多少気になりますがこのソフト無しで絵を描く苦労を考えれば仕方のないことでしょう。

色もれの対策としては，途中で何回かセーブすることをおすすめします。

そのためにもディスク版のほうが便利でしょう。テープ版ではセーブ・ロードに時間がかかります。

このソフト使用例として，おなじみのMZペンギンを描いてみました。（杉浦勇一）

①パソコンアニメーター



カラー23ページも見てね

次号予告

THE SOFTOUCH

SOFT TOUCHは，読者の皆さまからのご要望に応えて誌面を拡充し，市販ソフトの紹介にあたっては「新作ニュース」，「試用記」など，読者の皆さまがソフト購入のさいの参考になるよう配慮したいと考えます。「愛読者カード」に「紹介してほしいパッケージソフト」の欄を設けておりますので，特に誌面で検討を希望するソフト名（メーカー名）を記入していただければ幸甚です。

②ユーティリティ
③BASIC
④CZ-800C(グラフィックRAM要)
⑤カセットテープ
　ディスケット
⑥9,800円(ディスク用)
　5,800円(テープ用)
⑦㈱ストラットフォードコンピュ
　ータセンター・マジカルズゥ

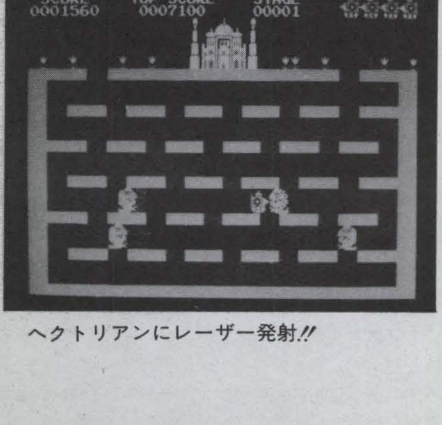
地底タンクに乗って イタサンドリアスを守れ

ハドソンソフト久々の新作ゲーム。同社のゲームというと以前は，オールBASIC記述のあまり面白くないものが多かったようですが，今回発売されたゲームソフトは，オール機械語。最近社会問題となりつつあるレンタルソフトを意識してか，モニタをも付けたIPLからロードする方式となっています。

ゲーム画面はカラー，表示する文字はすべてソフトで組んだもの(PCGをソフト上でグラフィックで処理した。最近テーブルゲームでよく見かけるATARI 社のゲームキャラジェネに似たもの)。デモンストレーションもなかなか凝っていて，これがかつてのハドソンかと思わせるほどソフト製作技術の向上がよくわかります。

唯一残念なことは，テーブルゲームのディグダグに似ていること。ゲーム内容そのものはかなり異なるようですが，画面といい，キャラクタなどが似てしまっています。ハドソンは以前にもNamco社のパックマンを参考にしたと思われるパックパックというゲームを発表していますから，オリジナル性という点ではこのイタサンドリアスも若干欠けていると思います。

①イタサンドリアス
②反射神経型ゲーム



ヘクトリアンにレーザー発射!!

③マシン語（IPLからロード）
④MZ-2200，MZ-2000(G-RAM I，2，3要)
⑤テープ
⑥3,600円
⑦ハドソンソフト

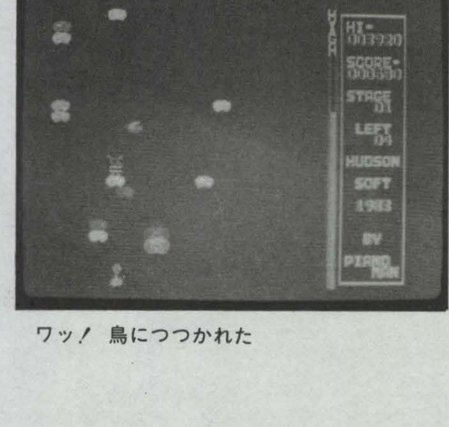
大空をひとりじめ 空には男のロマンがある!?

イタサンドリアス同様ハドソンソフトの新作ゲームの1本です。これもやはり，IPLからロードするタイプの機械語ですが，見ている限りでは，リアルタイムゲームであるにもかかわらず，あまり処理が高速であるとはいえません。このゲームはキー入力の反応が命であるにもかかわらず，どちらかというと画面処理を優先してしまっていることに疑問を感じます。

ゲームの内容は，オリジナルのようでストーリーも一応良くできています。飛行機から飛び降り，うまく地上に降下すれば良いのですが，空中でいろいろなものが飛んでいるため，紙ヒコーキを用いてそれらを防ぐというもの。マイコンゲームの中では最近，戦争ものの爆弾とか核兵器などがとびだし，過激の一途をたどっていますが，なかなか紙ヒコーキとは健全で思わずユニークさに微笑んでしまいました。

カラー画面の配色は，空色が強すぎて，キャラクタがあまりきれいに見えないのが残念。しかし，相当の開発時間を要したことは間違いないようです。

①スカイダイバー
②反射神経型ゲーム
③マシン語（IPLからロード）
④MZ-2200, MZ-2000(G-RAMI, 2, 3要)
⑤テープ
⑥3,200円
⑦ハドソンソフト



ワッ！ 鳥につつかれた

住所録＋汎用集計表 まさに，エコノミーパック

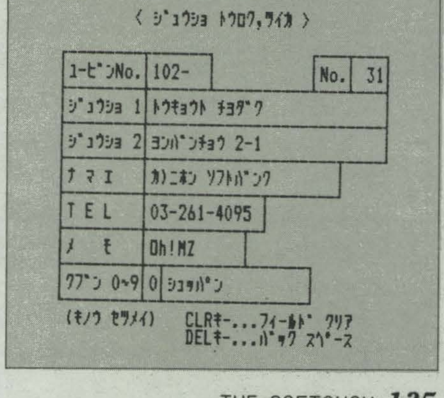
このパッケージは，「住所録」と「汎用集計表」という2本のプログラムを収めています。「住所録」は，区分登録，住所検策・修正・削除，ラベルプリント，住所登録および一覧表出力の4つの基本機能を持ち，最大240名分が一度に処理できます。プログラム自体オールBASIC記述で，初心者への誤操作に対する対策も万全で良くできていると思います。しかし，BASIC という制約から，処理速度は決して速くなく，若干のいらだちを覚えますが，リストを解析して改良を加えれば，十分実用にも耐えるものでしょう。

「汎用集計表」は，いわゆるデータベースの一種で先月紹介したS.CALGOのコンパクト版といったものです。表のサイズは，26×80で，そのうち任意の6×17をCRT上に表示します。命令は20あり，独特のコマンド群ですが，すぐに覚えられるほど簡素なものとなっています。モードはメインモード(データ入力)から2つに分れ，計3本のサブルーチンで処理しています。もちろん簡単な四則演算がサポートされていますから，日常使用にも耐えられると思います。今後バージョンアップをするならば，是非ともソートが欲しいものです。

以上，2本のプログラムに親切な使用説明書が付いていますので，初心者のかたにもお推めできるお買得パッケージです。

①エコノミーパックPART I
②ビジネス
③BASIC
④MZ-80B/2200/2000共用。
⑤テープ　⑥4,800円
⑦マイクロデータベースアソシエイツ

図2



MZ-5500へのMach 3フライトシ

シャープ初の16ビット機発売に伴い，PC-9801用に開発された「Mach 3フライトシミュレータ」が，MZ-5500に移植されました。

マシン発売前の作業ということもあって，ソフトのバージョンも定まらないうちの移植作業は，思いのほかたいへんだったようです。

まず，数少ないMZ-5500インプレッションも含め，移植作業をレポートしてもらいました。

0 はじめに

シャープからMZ-5500が発売されました。ハードウェアによるウィンドウ機能，マウス，16ビット機としてはめずらしいPSG, GDCによる高速描画，5インチフロッピー内蔵，CP/M-86の採用など，なかなかおもしろいパソコンです。

CPUが8086，グラフィック描画がGDCということで，NECのPC-9801用に作った「フライトシミュレータ」を移植しました。簡単にあがると思った移植作業が，ハード，ソフトの違いから結構手間どり，1週間弱を要してしまいました。コントロールをマウスでもできるようにし，ジェット音，爆発音もサポートしたもので，PCのシミュレータよりも一段と迫力あるものになったと思います。

移植上の問題点は，そのままPC-9801とMZ-5500との違い，長所短所となると思いますので，移植開始から終了までをレポートしてみたいと思います。

1 ソースファイルの転送

もとのフライトシミュレータは，PC-9801のCP/M-86上で開発したものです。ターゲットシステム上ですべての開発をしたほうが効率がよいと考え，PC-9801のCP/M-86上にあるアセンブリソースをMZ-5500に転送することにしました。

転送手段として，RS-232Cかメディア変換が思い浮かびます。232Cは安全確実な反面時間がかかるし，他社と形状の違うMZ-5500用の232Cコネクタやヌルモデムを新たに作るのもたいへんです。CP/M-86のツールも多く転送したかったため，メディア変換でやることにしました。できればコネクタなど他社と統一してほしかったと思います。

PC-9801上のソースファイルは，8インチフロッピーに入っていたので，IBMフォーマットに変換後，PC-8801のCP/Mで5インチに変換しました。

MZ-5500はMZ-2200との互換性をよく考えてあり，MZ-2000のCP/MファイルをドライブE,Fでサポートしているため，あとはPIP A：=F：*.*でめでたくアセンブリソースが転送できました。

また，SID86, Word Master, Word Starなどの開発ツールも転送しました。

2 ソースファイルの修正

アセンブリソースの修正はEDで行ないました。Word Masterなどのスクリーンエディタをインストールして使おうと考えていたのですが，MZ-5500のエスケープシーケンスが微妙に違い，難航しそうだったので，EDで修正することにしました（新バージョンのものはWard Starも動くようになっているので，スクリーンエディット可能）。

PC-9801上での開発時もやはりEDしかなく，それで行なったのですが，CP/Mでスクリーンエディタを使うのは，なかば常識といえます。エスケープシーケンスもきっちりソロクコンパチブルに合わせてほしいと思いました。

アセンブリ言語レベルの変更点は，I/Oポートアドレスの変更とPC-9801のROM内BIOSをコールしている部分をMZ-5500のIOCSコールに書き換えることでした。

I/Oポートアドレスの変更は，EQUを変えるだけだと思っていたのですが，GDCのコマンド，パラメータ出力ポートが102H，100Hだったため，

```
OUT  COMMAND_OUT_PORT, AL
```

は，

```
MOV  DX, COMMAND_OUT_PORT
OUT  DX, AL
```

と，DXレジスタ間接に書き換える必要がありました。8086は直接指定では0～255のポートしかアクセスできないためです。

また，MZ-5500のV-RAMのGDCからのアドレスは，標準でも

青　00000H～03FFFH
赤　08000H～0BFFFH
緑　10000H～13FFFH

となっています。GDCのパラメータ出力時にEAD（描画実行開始アドレス）を計算する必要がありますが，MZ-5500の場合，緑のプレーンを描画する際はEADが16ビットで表せなくなるので，この点も修正を加えました。

PC-9801のROM内BIOSをコールしているのは，

キー入力，キーボードイニシャライズの部分ですが，MZ-5500のIOCSはなかなかしっかりしていて，IOCSコールするだけで十分用がたせます。ちなみにIOCSコールの方法は，

キー入力（リアルタイムキー入力）

```
MOV   CX, 12   (コマンド)
INT   80       (IOCSコール)
```

（これでAXレジスタにキーコードが入る）

キーボードイニシャライズ

```
MOV   CX, 13 (コマンド)
INT   80
```

となっており，コマンド番号をCXレジスタに入れて，INT 80で行なっています。入出力パラメータは，少ない場合はレジスタで，フロッピー関係など多い場合は共有エリアを介して行なっています。

PC-9801上のソースファイルには，地形データなどBASICで作ったために入っていなかったのですが，MZ-5500はソースファイル中にDWで入れることにしました。

これでアセンブリソースの修正が終了したので，さっそくアセンブルしてデバッグに入ることにしました。

3 アセンブル&デバッグ

フライトシミュレータは，フライトのシミュレーション部が3Dパッケージをコールする形になっているため，3Dパッケージからデバッグしていきました。

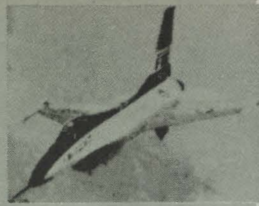
アセンブルはASM86を使いました。ASM86はマクロのないごく平凡なアセンブラで，Z-80などでアセンブリプログラムを書いたことのある人なら比較的簡単に移行できると思います。ただ，8086で本格的にアセンブルするには，少し物足りない感じです。

アセンブルは，いつも

```
ASM86  3D  $PZ
```

と，リスティングを抑えて行ないました。これは，MZ-5500のフロッピーが320Kバイトと容量が小さく，リスティングファイルを作るとすぐにいっぱいになるためと，アセンブル時間を短かくするためです。なにしろ320K

ミュレータの移植　田中　明夫



前後のアセンブルですから，全工程で10ぐらいかかってしまいます。

デバッグには，シンボリックデバッガのSID86を使いました。PC-9801上の開発のときはDDT86でデバッグしたのですが，やはりシンボリックでないと使いづらいようです。

デバッグといっても，PC-9801上で動いているのでほとんど必要ありません。ただ，MZ-5500はゼロデバイド時に，

INTERRUPT　TRAT　HALT

というメッセージを出してHALTしてしまうので，ゼロデバイドのトラップルーチンを書き換えました。

これで一応機械語部分は正常に動作するようになりました。次にBASICとのリンクです。

4 BASICとのリンク

なにしろ，これがいちばん手間どりました。

機械語ルーチンとBASICとのリンクは，マニュアルにも少し書かれていますが，それ以外に機械語は，

「8080モデル」でないと動かない。また，

「DSは自分でCSに合わせる必要がある」

そして，

「ORGは00H～0FHでなければならない」

「絶対アドレス指定でなければならない」

これらのことを守って初めて機械語ルーチンは正常動作します。機械語ルーチンとのリンクは，アプリケーション製作上，もっとも重要なことですから，もう少し詳しい説明があっても良いですね。SID86上では動くのにBASICとリンクしたら動かない，という状態が続き困りました。

BASICにモニタコマンドがないため，メモリダンプや変更，部分実行ができず，だいぶやりにくい思いをしました。メモリダンプなどはBASICでサポートすべきでしょう。また，メモリに落ちたものにパッチをあてても，それをセーブ，ロードするコマンドがないため，1バイトの変更にも再アセンブルする必要があり，たいへんでした。

なんとかBASICとリンクできたので，さっそくBASIC部分の移植を開始しました。

5 プログラムの移動

マイクロソフト系BASIC同志，あるいは同レベルのBASICの移植なら機械的にできるのですが，MZ-2000との互換性を重視するメーカーの姿勢があって，16ビット機のBASICとしては，いまいち貧弱と言わざるを得ません。気付いた点を正直に述べると，

●ラベルが使えない

のは当然として，

●変数名が2文字しか識別されない

のはかなりきつい制約です。プログラムのわかりやすさからいっても，せめて6文字はほしいところです。アセンブラでも16文字とか32文字識別のものが多いのですから。

●elseがない
●HEX$がない
●マシン語コールがUSRしかない
●音の出力がMUSIC文しかない
●メモリのセーブ，ロード機能がない
●モニタがない

こんな状態でしたがなんとか移植しました。BASICは，新しくBASIC-2が出るようなので，そちらに期待しましょう。

これでPC-9801と同等のフライトシミュレータができあがりました。

6 マウスのサポート

せっかくマウスが付くのだから，マウスでもコントロールできるようにしました。マウスもIOCSでサポートされています。

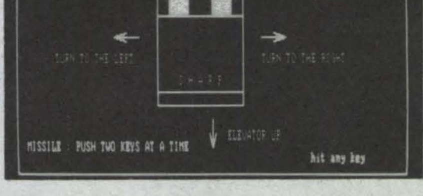
リアルタイムのマウスの入力は，

```
MOV  CX, 17    (コマンド)
INT  80        (IOCSコール)
```

で行なえます。データは，共通領域のオフセットアドレス3A2Hから3バイトに格納されています。共通領域のセグメントアドレスは，

```
MOV   CX, 1
INT   80
```

で，AXに入ってきます。これを利用して，



バンクとヘディングをコーディネートする形で，マウスによるコントロールを実現しました（写真参照）。

7 PSGのサポート

PSGが付いているので，ジェット音，爆発音，ミサイル発射音を出るようにしました。

PSGのレジスタの設定はIOCSにはないので，直接I/Oポートに送りました。PSGの各レジスタにデータをセットするには，

```
MOV  AL, レジスタ番号 ┐
MOV  DX, 231H         ├ PSGのレジスタの指定
OUT  DX, AL           ┘

MOV  AL, 出力データ   ┐
MOV  DX, 230          ├ データの出力
OUT  DX, AL           ┘
```

で行ないます。やはり，音があるとないとでは，かなり迫力が違いますね。

8 デモ画面の取り込み

カラーページの写真を見てください。このフライトシミュレータには，F16のデモ画面がついています。ときどき「よく拾いましたね。あれだけのデータを」といわれますが，あれはテレビカメラからの画像をJUNの4Dでビデオデジタイズしたものです。

これをPC-9801のディスクから，メディア変換で吸いあげて，移植作業は終了しました。

回 おわりに

思わぬハード，ソフトの違いで，1週間弱を用した作業でしたが，MZ-5500のハード，ソフトを肌で感じられ，なかなか有意義な作業だったと思っています。

MZ-5500は，ウィンドウ機能やマウス，PSGなど，ハードがすばらしい反面，それをサポートするソフトがいまいち強力だとはいえません。たとえば，マウスはBASICからではMSINコマンドでしかサポートされません。ウィンドウもBASICコマンドによるサポートだけではもったいないでしょう。コンカレントCP/M-86で各タスクに各ウィンドウを割り当てるなどおもしろいと思うのですが。

BASIC-2　をはじめとするMZ-5500のハードを最大限に利用したソフトの出現に期待したいと思います。

STUDIO mz

◆SHARPさん，〝MＺも部屋の一部です〞カラーオーダーやりませんか！

林　弘和（16）神奈川県

ぼくの部屋に，パステルカラーのMＺ！　な～んていったりして…。

◆どうして，このOh！MZはこんなにも魅力的なのでしょう。大学に落ちたら，責任をとってもらいますヨ！　久保　満（17）福井県

まかせなさい！〝大学へのOh！MZ〞創刊!?

◆MZ-700のユーザーです。今，マシン語を勉強していますが，LOGOなどという言葉が世に出回りはじめ，頭がいたい今日このごろです。しかしOh！MZがあればだいじょうぶ！なんとOh！MZは頭痛薬にもなるのです。

磯崎　保彦（？）神奈川県

書店で売り切れのときは，薬局へも行ってみよう。

◆クリーン思想も良し悪しで，電源ONですぐに計算ができず，暗算が強くなった今日このごろです。　山端　恒和（28）大阪府

おそれいります。

◆日本ソフトバンクというと，愛人バンクとか，血液バンクとか友人にやじられてこまる。MZは，マジンガーＺの略であるといううわさは本当ですか。　飯島　徹（13）福井県

ちなみに，日本ソフトバンクを略すと，〝日本そば〞になることは知っていましたか？

◆この前，友人が自分の声でコンピュータの音と同じような声を出し，録音してMZにかけていた。結果は当然ダメ。理由を聞いてみると，自分でソフトを買う金がないので，何かおもしろいものでもできるかと思ってやったそうだ。このアホ!!

中村　明彦（18）愛知県

気持はわかりますが…。やっぱりアホ!!

MZ-700にフロッピーを!!

◆9月号の皆川君への「…MZ-700に標準フロッピーをつけてCP/Mマシンに昇格させよう」とありますが，シャープさんに出すのか出さないのか，MZ-700のユーザーの代表質問をして，いつ頃になるか11月号に載せてほしい。前にもフロッピー用の端子の話が写真入りであったり，MZ-700関係の本にも載っていた。これではへびの生殺しで買い替え族も増えるのでは!?　まさかそこがネライではないでしょうが。　山内　道雄（43）埼玉県

◆MZ-700シリーズ用のフロッピーはいつ出るのでしょうかねェ～。首を長くして待っているうちに，ロクロ首になりそーです。

鈴木　道隆（31）兵庫県

◆「ディスクBASICをあつかう法」」を読んでMZ-700用のフロッピーが待ちどおしくなりました。どうか，シャープにさぐりを入れてください。

藤井　伸二（22）神奈川県

◆今度SHARPさんへ，700のフロッピーはいつでるのか聞いてください。マニュアルには発売予定と書いてあるのに…。　清水　隆弘（14）愛知県

◆SHARPの社長さん，MZ-700のフロッピーを作って！　森岡　哲矢（12）岐阜県

◆本物のワープロがほしい。MZ-700にディスクを！　漢字プリンターを！　I/Oポートを！　……夢かな？　それとも，マシンを取り替えようかな？　でも金がない！　金子　宏（28）埼玉県

◆MZ-700は周辺機器が少ない!!　早くフロッピーディスクが出ないかなあ…。

平林　毅（14）長野県

ユーザーあってのMZです。SHARPさ～ん。おねがいしますよ。

厚切りパンのようにもっと厚くな～れ、Oh！MZ。MZ-700のS-BASICで、〝TROFF〞〝TRON〞の使用ができるヨ！
三星　俊則（27）千葉県

こうもと・やすひこさんが最高！　理解しやすかった。
梨本　美樹（18）東京都

◆「時をかけるバグ」

「あなた～わたしの元から～とつぜん暴走しーないでねぇ～え　二度とはつかえないプログラムにならないとちかってぇ～」　どうも失礼しました。みなさん，バグには気をつけましょう！

酒井　康成（13）兵庫県

◆ぼくは今までPCファンでしたが，クリーン設計にあこれ，MZファンになりました。コンポタイプにルンルン気分です。

生駒　龍司（14）大阪府

◆今年の冬，パソコンというものを知った，ごく初心者です。3月に高校合格でMZ-731を買いました。Oh！MZは6月号から毎回買っています。初心者向きの記事もあり，内容も充実していると思います。BASICはひととおりマスターしたつもりですが，マシン語については全然わかりません。バリバリの進学校でいそがしいのですが，長くおつきあいしたいと思ってまっす!!

岡田　伊知郎（16）岡山県

◆今あるマイコン雑誌の中で，一番読みやすくてMZのことがいっぱいのっているのでとても気にいっています。ハード入門がとてもいいのでこれからもつづけてください。松下　進（17）京都府

◆もっと記事を増やして，プログラムをいっぱい載せて!!　そうすれば本が厚くなる。ぼくは，厚い本のほうが得したようで好きです。

菊地　大作（16）千葉県

◆最近，私の妻はMZ-700の始動音を聞くと悪寒が走るそうです……。しかたなく，私は今夜も妻が寝つくのを待っています。

木下　芳男（26）福島県

◆MZ-700を買ったばかりのころは，ゲームばかりしていて，ルンルン♡でしたが，最近ふとこれでよいのだろうかという思いが…。周辺機器なんてchinpun kanpun。あーちくしょー酒もってこ～い!!　望月　直美（17）静岡県

◆この前，本屋でOh！MZを取ろうとしたら，となりにいたおじさんがジロリとにらむので，早くレジのとこに行こうとした…とその時，そのおっさんが「ばかヤロー!!」と大声でぼくにむかってさけんでいた。ぼくはOh！MZを買うのがこわい（それから，そのおっさんはOh！PCを読んどったぞ!!）。　青木　宏整（14）東京都

◆最近，巨大なおばけBASICがはやってますね。先祖返り気味の僕はMZ-700にTiny BASICを移植したいと思う今日このごろ。シャープから（他でもいいから）整数BASICインタリプリタ（4～8K byte）が出たら感動!!……なのにな。ああ，2K BASIC!!　松原　隆行（16）千葉県

◆プログラムがたくさんのっていて，その解説をしたり，ハードの勉強ができたり，とにかくいい本ですねェー。特に9月号は表紙がすごい。めちゃんこすごい。　清水　秀人（17）秋田県

◆現在のOh！MZは後世の人々に残してあげたいですね。しかし酸性紙の寿命は100年末満，もっともっと長持ちする中性紙があるけど，Oh！MZはどっちの紙？　渡辺　弘（15）宮城県

編集部から読者の皆さんへ
お便りをお寄せください。内容はどんなものでもかまいません。言いたい放題、ユニークな内容のものをお待ちしています。あわせてイラストも大募集。STUDIO MZまでどんどんお送りください。

◆ELDER MANさんのドライブゲームをジョイスティック化して楽しんでいます。128秒で228.6kmでした。それとDATA文を変えて自分のコースをつくりました。新連載が楽しみ。

三浦　剛 (17) 千葉県

今月号からのボーリングも楽しんでいただけるでしょう。

◆最近，マニュアルを読み返してみました。すると前にはチンプンカンプンだったことが今はチンプンカくらいでした。他の人にくらべるとナメクジみたいにおそ〜い進歩だけれども，やっぱり進んでいるんだなぁ！　と実感する今日このごろでした。

浜崎　健一 (20) 広島市

今月号でチンプンくらいになったかナ？

◆はじめて買いました。他のマイコン誌とちがいMZのことばかりでよいと思いました。「英文ワープロ」(9月号)をやろうと思いますが，機械語を入れるのはしんどいです。

中尾　雄一 (17) 京都府

9月号の大津良之さんのプログラムは，リストを打ち込むのは大変ですが，完成させれば値千金の強力な英文ワープロです。あきらめずに挑戦しよう。

◆8月号「特集LOGO新しい時代の言語」，9月号の「入門タートルグラフィック」を読んで大いに感動しました。今後もこういう記事を期待するMZ-2000少年がいますのでがんばってください。

田中　恵介 (17) 兵庫県

MZ-LOGOは生まれたばかりの言語です。新しい可能性を引きだすのは，読者のみなさんの手にかかっているのですヨ。

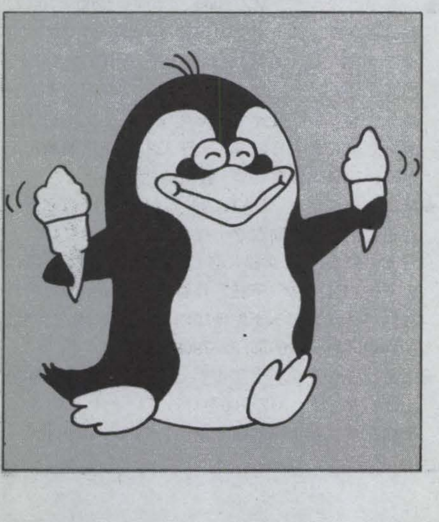
◆先日，X1のキーボード延長コードを買った。何をかくそう，実はステレオのミニヘッドホン用の代用で850円。おかげで長時間にらめっこしていてもあまり疲れなくなった。しかし，テープの出し入れが不便になった。マジックハンドでもあれば，唯一手動であったカセットぶたを閉めることもできるようになるだろうか……。

石崎　章司 (21) 千葉県

キーボードを膝の上にしてゆったりとソファーに腰しかける。そしておもむろにキーを……。優雅だなあ！

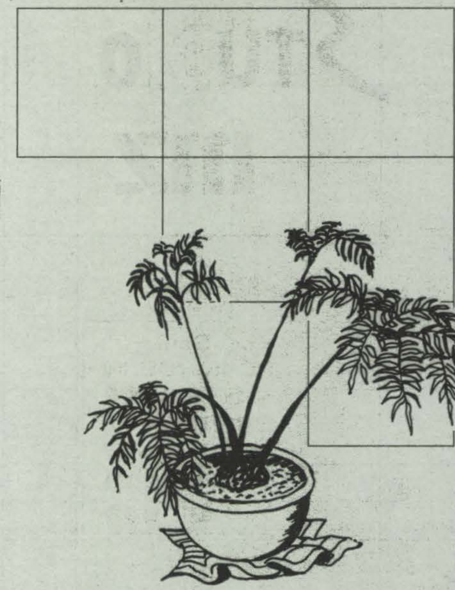
◆9月号の関亦康雄君に賛成。キレイなおねえさんを復活させよう！　なぜ中年のおじさんは，あれをきらうのだろう？　藤岡　明彦 (16) 岡山県

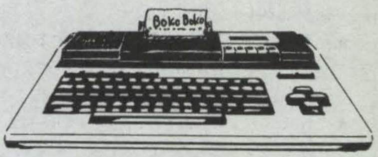
別にきらっているわけではないのです。中年のおじさんは，好きなものはこっそり見たいのです。



私のMZ-700は ボコボコという名前です。
BOKOは ノート以外は うまく絵が描けません。頭の中に描けないのです。なんとかしないと… バカバカになっちゃう。。。
この前 プログラムを入力してたの。雨の日でカミナリくんが 叫んでた。ピカッ!!と光って急に停電になっちゃった。Ah— もうすぐ終わりだったのにィー。カミナリくんのばかぁ!!

田中祐美 (14才)　Addie S58.9.11.





◆「某社の製品のように」とか，「他社の機械（とくにN社の88ナントカ)にみせてやりたい。」などの文章は大変良いのでもっと多く載せて欲しい。MZ専門誌なので他機種をもっとけなそうー

笹原　章嗣 (28) 兵庫県

私の席は，Oh！PCのセイコちゃんと向いあわせ，Oh！FMのマリちゃんと背中あわせ。私が挑戦的な態度を見せると，不敵な笑みが帰って来ます。まけるもんか！

◆MZ-700のOWNER'S MANUALなんですが，友人のにはBASICのBACK UP COPYの方法が書いてあるのに，ぼくのには載っていません。どうしてでしょう。　国枝　昌彦 (15) 大阪府

だから，Oh！MZをよろしく！

◆ぼくは，MZ-731を戦争用にカムフラージュ色にぬった。ざまみろ！　9月号P.130の井上君，まいったか！　片山　章 (13) 山口県

MZもカラーコーディネートの時代か？

◆前線基地について一言，キーボードがこわれる。

有吉　和美 (?) 福岡県

ちょっと，熱くなりすぎでは？

◆ぼくは，6月号のアンケートで，8月号に「読売から編集室」になっていると指摘した者です。もしかして，名前が「STUDIO MZ」にかわったのは，ぼくのせいなのでしょうか。

妹尾　高史 (13) 大阪府

よくわかりましたね，そう，キミのせいなんですよ。フッフッフ。

◆MZ-80K/Cなんて，ぜんぜん耳にしない今日このごろ。新機種ラッシュがはげしくなり，MZ-2000でさえ12万円で売られている。ジョークでいい。表紙にMZ-80K2Eをのせてくれ！

吉田　宏士 (13) 兵庫県

マシン室のスミでは，まだまだMZ-K2もガンバッテいるのです。

◆9月号は大満足！　2000を手放さなくてよかったぜ！　2000を売っていたら，今頃…？　今度X1も買ったので2000のプログラム同様X1のプログラムもたくさん紹介してくださいネ。よろしく！

田打　幸弘 (17) 滋賀県

MZ-2000とX1！　うらやましい〜。

◆ゲームを作っていると，構想ではすっごく面白いゲームになるはずが，実際につくってみると…ということがよくあります。理想にどれだけ近づけられるかということもプログラミングの上で頭の痛くなる問題のひとつだと思います。現在PCGを使ったゲームを構想中。スピーディーでロマンの香りだようすっごく面白いゲームになるはずなのですが……。あ，もちオール機械語。となるとやっぱりシステムプログラム欲しいなあ。

冨樫　道生 (22) 東京都

期待してますヨ！　完成したら投稿してネ！

◆な，なんで「MZ-通信」のハガキが消えてしまったんだ!!　郵便局で売ってるハガキなんかきらいだー。　神田　倫明 (14) 東京都

あのハガキは私がDesignしたのに…，グスッ。

◆つ，ついに買ってしまった!!　MZ-2200。ついでにカラーディスプレイとデータレコーダーも買った。9月号の田中さん，なんだなんだMZ-2000ってMZ-2200のほうがいいんだい！

熊谷　聡 (12) 島根県

◆9月号で「プログラムを拡大する物を発明して…」というのがありましたが製品として売っています。アドコム電子から，リストルーペという名で，大きさは大中小あり，それぞれ4800，2300，900円です。使っていますが，なかなか便利です。

小川　裕司 (16) 栃木県

◆"読者から編集室へ"から"STUDIO MZ"とかわりましたねー！　こういうデザインはI Love！　Dr. Kuのおじさーん！　こんどはCZをお願いしますー。　豊島　隆之 (14) 長崎長

◆なぜ，もっと早くこんな本があることを教えてくれなかったんです（知ってはいたけど)。おかげでぼくは，7月号から買いはじめたばかり。聞くところによると，「マシンゴチョーニューモン」などあったらしいではないですか！　クソー。

近藤　顕治 (15) 大阪府

◆バックナンバーフェアをしてください。

中村　友通 (14) 京都府

現在，在庫のあるバックナンバーは，'82／10〜12, '83／2，5〜10月号です。最寄りの書店にお申し込みください。

STUDIO MZ

For Sale
PC-1500+CE-150を4万5千円で。関連図書2冊付。本年7月購入。　〒157 東京都世田谷区成城5-8-3 高井　しゅう

売ります

★MZ-721。新品同様＋ロータスゲームボード＋ゲームソフト多数＋関連図書＋カバーを6万円ほどで。少々なら応談しますヨ。まずは往復ハガキをください。〒836 福岡県大牟田市新地町4の2 守瀬　博人

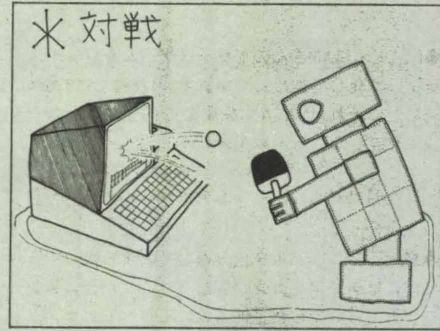
★私めの可愛いMZ-731をお譲りします。3ヶ月使用。美品です。ロールペーパー，ボールペン新品です。8万円で。送料こちら持ち。W〒待ちます。
〒010 秋田市手形田中5-2 石郷岡アパート16号 小沢　宏明

★MZ-1200（本2冊，ゲームソフト6巻，BASIC1巻）を7万円で売ります。送料はこちらもちです。希望の方は，往復はがきで。
〒370 群馬県高崎市台町35　磯部　篤

★MZ-2000＋G-RAM1,2,3＋Hu-BASIC＋拡張MZ-1Z001＋書籍類（保，59年6月迄箱付）＋α以上を15万円で。〒187 東京都小平市学園西町1-29-1一橋寮北204号　田井　公一

★売ります！　MZ-80Bを2000にするMZ-2000コンパチボードSID1001を2万5千円で。
〒805北九州市八幡東区宮田町19-25　西岡　介士

★売ります。PC-1211（本体）＋CE-122（プリンタ／カセットインターフェイス）＋その他を1万5千で!!まずは手紙で!!　〒470-01 愛知県愛知郡東郷町大字諸輪字北山111-336　谷　俊宏



TOMUC 岡田 進

★SHARP PC-1251とCE125と付属品一式＋ソフト数本を3万5千円で（完動品）。送料はこちらで持ちます。価格相談可。
〒580 大阪府松原市小川町9-43　岡本　隆宏

★売る!! I/Oデータ機器PIO-3055（漢字ROMボード）80B／2000用を2万5千円で（ワープロminiを付けます）。早い者勝ち。TEL乞う。
☎0569-23-1917　谷川　清彦

買います

求む。MZ-700用カラープロッタプリンタを1万円にて。完動品のみ。W〒待つ。〒437-13 静岡県小笠郡大須賀町沖之須283　大石　勝男

★MZ-2000＋G-RAM1,2,3＋カラーBASICを9万円で。拡張ユニット・システムソフト付は10万円。または，MZ-731を4万円で。W〒ください。完動・無改造であればキズ可。
〒489 愛知県瀬戸市萩山台2-124　磯部　六明

★MZ-2000用のカラーモニタ，ライトペン，フロッピーなど，なんでもいいです。まずはW〒で。
〒311-17 茨城県行方郡北浦村繁畠46　辺田　政寛

★MZ-8BK＋MZ-8BGKを2万円以下でおねがいします。W〒で連絡を！
〒543 大阪市天王寺区上本町3-3-8　伊藤　直樹

★MZ-700用カラープロッタプリンタ2万円ぐらいで売って！　〒731-15 広島県山県郡千代田町下十日市489-1　立川　清

★求む！　CZ-800P（プリンタ）を7万円くらいで，完動ならキズ可。W〒待つ。
〒579 大阪府東大阪市末広町15-9　辰己　忠資

★X1（CZ-800C，CZ-800D）を買います。価格応談，手渡しを望む。
〒446 安城市上条町西荒井1　阿部　建夫

★X1のG-RAM（CZ-8GR）を2万円ぐらいで。
〒081 北海道上川郡新得町栄町　松坂　賢一

★Oh！MZ創刊号〜9月号までと，1月号を各600円で買いたいのでお願いします。　〒593 大阪府堺市菱木128 福泉住宅2894　細谷　利明

★買います。MZ-80FIO（フロッピーI/Oカード）を12万円で求む。まずはW〒で。
〒703 岡山市円山99-7　江尻　泰幸

交　換

★当方MZ-2000＋G1,2,3＋ソフト計35万以上の品。貴方バイク（RZ-250，350など）その他の車種も考える。または10万〜12万円で売る。まずはW〒で。〒571 大阪府門真市深田町18-3　大草　豊久

★当方MZ-731＋ソフト，貴方PC-1500＋CE-150＋周辺機器と。条件等，相談に応じます。
〒510 四日市川島町1684-29　小垣内　学

★当方MZ-1200＋ロータスゲームボードGB-10＋JOYスティック＋MANUAL，貴方MZ-700と交換してくれる方TEL待ってます。ハガキでもOK。
☎0886-63-0029　〒770 徳島市論田町小論田12-5 平島　英治

★当方MZ-731＋1万円，貴方PC-8201。W〒待つ。
〒564 摂津市庄屋1-1-17　長井　始

仲　間

★MZ-1200をお持ちの小6〜中2ぐらいの友達を求めています。いろいろなことを教えてくれる人ならよけいによいと思います。全国からのハガキ，手紙をお待ちしております。
〒830 福岡県久留米市荘島町20-2　森山　茂栄

★X1ユーザーの18才のぼくです。情報交換，プログラム作りを一緒にしましょう！　W〒，お手紙お待ちしています。　〒674 兵庫県明石市大久保町松陰新田703　鶴谷　佳尚

★全国のMZ-2000／2200のファンのみなさん！　ひとつ提案があります。第4日曜日午後3時，近くのパソコンショップに行って，ファンの集会をして団結しよう。この意見に関するお便りをW〒で待っています。「MZ-2200ノススメ」を書いて返信したいと思ってます。　〒651-11 兵庫県神戸市北区泉台2-7-12　大浦　善人

★全世界の皆さま，MZ-2000／2200や1200／700などいろいろなシャープのMZユーザーの方々，当クラブ"JASMUC"（Japan Sharp MICOM User's Club）に入ってください。詳しくは60円切手同封のフウトウかTELでお願いします（なお，Oh！MZが出てから2ヶ月のみ）。　〒338 埼玉県浦和市上木崎4-1-34　皆合　努　☎0488-31-7958

★マイコンクラブ"DATA IN SASEBO"設立!!　初心者・女性大歓迎!!　現在，会員の中には，メーカーの人，マシン語に詳しい人がいます。連絡はW〒かTELで!!　〒857 長崎県佐世保市潮見町16-11 パソコンプラザ佐世保内「DATA IN SASEBO事務局」　☎0956-32-7908

★文通しましょう。小学4年で，ベーシックを始めたばかりです。MZ-700を待っています。
〒157 東京都世田谷区北烏山1-59-8　森　徹

◆掲載ご希望の方は，官製ハガキに項目（売る・求む……）を明記してお申し込みください。
◆ソフトの売買，交換については，いっさい掲載できません。
◆取り引きについては当編集部では責任を負いかねます。
◆応募者多数の場合，掲載できない場合もあります。

㈱日本ソフトバンク発行のOh!シリーズ

月刊

Oh!PC



好評発売中！

11月号　定価480円

■特集
ワードプロセッサ・テストランレポート

- カラーグラフ秋の新製品　PC-9801F
- 大型オリジナルソフト
  PC-9801家計簿プログラム
- PC-8801でビデオ画面を出力してみよう！
  [ゲームプログラム] PC-8201四人麻雀
  好評連載：DISK CHARGE/Softload
  わかっていいとも/機械語熟語集

季刊

Oh!HC



好評発売中！

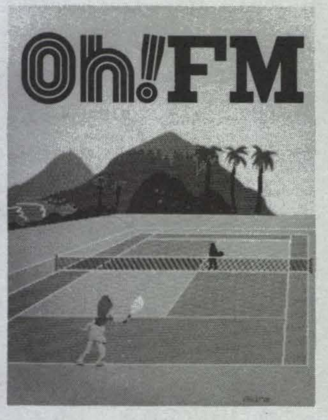
第5号　定価480円

■特集・**HC-20完全マスター**
やさしいEPSON BASIC
気軽に使おうサブルーチン
ペリフェラルで機能拡張

- 商店経営診断シリーズ　販売効率分析
- 入門シリーズ　プログラム開発手法
- 実験レポート　ACコントローラ
- 簡易言語と実務（1）
- 創刊1周年記念ビッグプレゼント

隔月刊

Oh!FM



好評発売中！

（増ページ）
第5号　定価480円

■特集　**グラフィックツールいろいろ**
グラフィックコマンドを使ってみよう/拡大・縮小グラフィックエディタ/グラフ作成ツールなど
付録：レイアウト/パターン　デザインシート

- OS-9V1.2とBASIC09
- はじめて握るはんだごて
  ジョイスティックの製作
- ゲームリスト　エイリアンFMクラッシュ

季刊

Oh!55



近日発売予定！

第2号　定価480円

■特集　**5550用パッケージソフト**

- IBM PC(5150)で5550の可能性を探る
  DR Logo
- 逆ポーランド記法実習プログラム
- MS-DOS用スクリーンエディタ
- マクロアセンブラ
- ケーススタディ　5550導入オフィス
- プログラム　バイオリズム相性率

COVER STORY〈8〉



R・ディレーニィの「バベル－17」は，ヒロインが詩人で美女，相手役も宇宙海賊，異星人etc…おなじみのスペースオペラもたっぷり楽しめるけれど，タイトルにもなっている異星人の言語，バベル－17に注目したい。なにしろ，ôX，たったこれだけで円(◯)の大小（>）と交差(X)関係などの数学概念を正確に表現できるような完璧な論理構造を持っている。そして，あなたや私にあたる単語がない。主述の関係がないのだ。

物語は，ヒロインのリドラが，バベル－17の秘密を探るうちに，逆にこの言語に魅了されて，ついには異星人にバベル－17で洗脳された男と，人体が融合する！という場面でクライマックスをむかえる。自と他の区別をなくした2人の精神を分離させるのに利用するパラドックス（矛盾律）がもの足りないという人もいるかもしれない。

さて，昔ハッカー，今シリコンバレイガイ。AIDSと並んで男性誌P誌に掲載されたコンピュータ依存症候群の原因は，プログラミング言語にあるのかもしれない。リドラも述べている，「バベル－17は，アルゴル，フォートラン，古代の人工語に似てるの…」

（誠子）

Puzzle BOX 11

ここは，良妻賢母教育で定評のある花野女子校。ルンルン組の生徒は，今日の家庭科の授業でビーズの腕輪を作ることになった。それは，赤，黄，緑の3色のビーズを等間隔で糸に通した簡単なものである。

先生は「ビーズは全部で7個使いなさい。色は，たとえば赤3個，黄2個，緑2個というように，どれかひとつの色を3個，残りの2色は2個ずつにすること」と指示した。

たまたま，生徒の人数が作れる組み合わせの数と同じだったので，生徒全員が他の人と違う腕輪を作ることができた。

そこで，問題。さて，ルンルン組の生徒は全部で何人いたのだろう。

応募方法

○官製ハガキに，答と住所・氏名・年齢を明記のうえ，
〒102 東京都千代田区四番町2－1
日本ソフトバンク　Oh！MZ PUZZLE BOX係
までお送りください。
○締め切り　11月15日到着分　○正解発表　1月号誌上
○正解者多数の場合，抽選で10名の方にOh！MZ特製記念品を差し上げます。

9月号クイズの正解

答　ジョージ

それぞれの生存確率　ジョージ$\frac{47}{90}$　スミス$\frac{3}{10}$　ブラウン$\frac{8}{45}$

三角様式のピストル決闘では，射撃の一番へたなジョージをおいて，まずスミスとブラウンがお互いにねらい合うだろう。

ブラウンとの射ち合いでスミスが，$\frac{1}{2}$の確率で先手となった場合，必ず射殺する。ブラウンが先手となった場合は，彼の命中率が$\frac{4}{5}$であることさら，このときのスミスの生存確率は$\frac{1}{5}$となる。したがって，スミスのブラウンに対する生存確率は

$\frac{1}{2}+\frac{1}{2}\times\frac{1}{5}=\frac{3}{5}$　となる。

次にスミスは，命中率$\frac{1}{2}$のジョージに対して$\frac{1}{2}$の生存率をもつ。したがって，スミスの結局の生存確率は

$\frac{3}{5}\times\frac{1}{2}=\frac{3}{10}$　となる。

ブラウンのスミスに対する生存率は$\frac{2}{5}$である。次に彼がジョージを殺す確率は　$\frac{1}{2}\times\frac{4}{5}=\frac{4}{10}$　である。しかし，ブラウンが失敗する確率は$\frac{1}{5}$であり，彼の生存率は$\frac{1}{2}$であるから，失敗した場合，$\frac{1}{2}\times\frac{1}{5}\times\frac{1}{2}\times\frac{4}{5}=\frac{4}{100}$ となり，さらに失敗していくと

$\frac{4}{10}+\frac{4}{100}+\frac{4}{10}+\frac{4}{10000}+\cdots$　となって，この数列の和は 0.44444…の循環無限小数。つまり$\frac{4}{9}$の小数展開である。したがって，ブラウンの結局の生存確率は

$\frac{2}{5}\times\frac{4}{9}=\frac{8}{45}$　となる。

ジョージの生存確率は，同様にわかるが，スミスとブラウンの生存確率を1から差し引くことで直接求められる。

$1-\frac{3}{10}-\frac{8}{45}=\frac{47}{90}$　となる。

正解者

（埼玉県）斉藤哲　（京都府）森谷英之様ほか8名の方に記念品を送らせていただきます。

ごめんなさいのページ

I'm Sorry.

9月号 MZ-LOGO

P.92 リスト5 rasen

2行目と3行目の間に fd:d を入れる

P.93 リスト7 design

1行目 to design1 :a :a→to design1 :d :a

3行目 dia :a → dia :d

P.94 8行目

3…ピンク→マゼンタ 5…マゼンタ→水色

P.94 リスト10 4行目の endを改行する

10月号 駆逐艦 秋風

P.34 510行 バグではありませんが，INKEYと$の間に0を入れるとキー操作が速くなります。

P.37 2740行 ……3120,310,3200→…3120,3160,3200

10月号 MZ-LOGOとリスト処理

P.107 6行目 output last 1 ………… 1字空ける

⑤の20行目 print length[A B C] 1字空ける

22行目 print length[A[B C]] 1字空ける

P.108 右段11行目 …… "a[a][a b c] 1字空ける

右段13行目 …… "X[[a]b a] 1字空ける

右段最終行 *dc*→*bc*

P.109 左段図7下の行

A＜B(積)→A∪B(和)　A＜B(和)→A∩B(積)

P.110 右段15行目

Print powerset [A B C] 1字空ける

[*A B C*][*A B*][*A C*][*B C*][*A*][*B*][*C*][ ] 1字空ける

P.111 左段17行目

Print makeset[a b c b d a] 1字空ける

右段15行目

match[a b c][a b c] 1字空ける

右段17行目

match[a b c][a b c] 1字空ける

右段図16の4行目

if equalp :com"f run, rec [fd 10]”

P.112 右段下から8行目

Print:bc → Print:b :c

右段下から6行目

match[?l b ?r][a b c d] 1字ずつ空ける

P.113 図24 5行目と7行目

"@ 間をつめる

10月号 Oh! MZ質問箱

P.166 中段下から3行目 MZ-8B103→MZ-8BI03

右段6行目 USTAT→USART

おことわり ▷空丹遊歩さんの「AROUND THE BASIC」は，著者にわかに開発の仕事がたてこんで，今回は残念ながらお休みです。また「絵夢絶登面白玉手箱」も遅ればせながらの〝夏休み〟で休載(有田クン，読者の皆さんが待ってますよ)です。▷ときおり小誌の広告ページから「資料請求券」を切り抜いて編集室あてに送ってくる方がいます。また，ときおり小誌の広告ページをみた，と現金同封でソフトパッケージを注文なさる方もいます。ありがたいことですが，いずれも編集室では処理できかねます。広告スポンサーあてにご送付ください。 (ja)

本誌のバグに関するお問い合わせは
受付電話 03(265)5789
受付時間 月～金曜日 16：00～18：00

なお，お問い合わせは，原則として本誌バグ情報のみに限らせていただいております。また，ご質問いただくときは，○月号○ページ，タイトルを明示してください。

2ヵ月前までのバグに関しては，このページでサポートしています。バックナンバーでお確めください。

SHARP MZ Software Information

(シャープ株式会社MZソフトウェアインフォメーションNo.4より掲載)

●シリアルインターフェースカードMZ-8BI03の取扱説明書(RS-232C SERIAL INTERFACE)に，下記内容の誤りがありましたので，訂正いたします。

P18 6．プログラミング

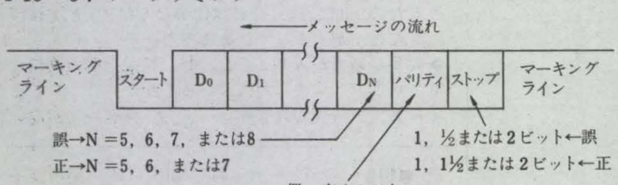


図6―1 非同期モードのメッセージフォーマット

●パーソナルコンピュータMZ-700の取扱説明書に，下記内容の誤りがありましたので，訂正いたします。

① 29ページ【コントロールコード表】

CTRL＋	アスキーコード(10進)	処理内容
C	3	ひらがなモードとする。(CTRL＋カナ)

▷上記CTRL＋Cの機能はありません。

この項は，抹消してください。

② 41ページ 下から13行目

誤 51．123E＋13……………

正 5．112E＋14……………

PRINT USING〝##．###↑↑↑↑〟のUSING文中の最初の#は，正負の判断に使用されます。

●パーソナルコンピュータMZ-80Bのシステムソフトウェア FLOPPY DOS内のBASICコンパイラの使用について。

〔内 容〕 指定ディスクにライトプロテクトシールが貼られている場合でも，OPENしようとするファイル名と同一のファイル名がすでに存在する場合には，〝WRITE PROTECTED〟のエラー表示が現れません。

また，ファイル内容の書きかえ動作も行なわれません。

〔プログラム例〕

```
XOPEN #1,"XYZ"
? #1(1),"AAAAAA"
CLOSE : END
```

(実行前にすでにディスク内に〝XYZ〟のファイルが存在)

3行目のCLOSEをCLOSE#1にするとエラー表示をします。

○編集室から

12月号――11月18日(金)発売

●特集　スモールビジネスプログラム

●X1／MZ-700マシン語詳解

ワープロ／ゲーム プログラムを大募集

多数のご投稿，いつもありがとうございます。今回のプログラム募集テーマは，ワープロ／ゲームプログラムです。

Oh！MZの読者の方の中には，プライベートなワープロを作って利用している方が多くいるのではないかと思います。本誌では近くワープロ特集を予定しています。皆さんの苦心作をお寄せください。

また，シミュレーション，アドベンチャー，ロールプレイングを中心としたゲームプログラムも募集します。ユニークな内容のものをお待ちしてます。

締め切りは，一応12月末としますが，投稿原稿については，随時お受けしています。どしどしお寄せください。なお，本誌掲載分には規定の原稿料をお支払いします。

●応募要領●

○内容の説明，利用法，フローチャートなどの原稿に，テープ（ディスケット）を添えてお送りください。原稿には住所・氏名・年齢・職業・マイコン歴を明記願います。

○投稿者のモラルとして，他誌との二重原稿，他機種用プログラムを単に移植したものは，固くお断わりいたします。また，プログラムのSAVEは，最低2回以上行なってください。

●本誌のお買い求めについて●

最近，本誌の品切れが相次ぎ，読者の皆様方にご迷惑をおかけしていることを深くお詫び致します。本誌は全国の大型書店，マイコンショップで販売しておりますので，予約ご購入が入手確実です。品切れのため，やむを得ず当社にご注文いただく場合は，送料実費をいただくことになります。

ご送金は郵便振替（東京1-29307）か現金書留で，ご注文書には住所，氏名と雑誌名何月号かを明記してください。なお，バックナンバー在庫等のお問い合わせは，出版部（00-261-4095）宛にお願い致します。

広告目次

Oh!MZ 11月号
◆1983年11月1日発行　定価480円
◆発行人　孫　正義
◆編集人　田鎖洋治郎
◆発売元　㈱日本ソフトバンク
出版部　〒102 東京都千代田区四番町2-1　☎03(261)4095　FAX03(262)8397
本社　〒102 東京都千代田区九段南2-3-14　靖国九段南ビル☎03(263)3690㈹
TELEX 東京232-4614　JSBTY
大阪支店　〒542 大阪市南区難波千日前5-19 河原センタービル3F
☎06(644)0191㈹　FAX 06(644)0160
◆印刷　凸版印刷株式会社

SHIFT BREAK

▶待望の本格的16ビットマシン。MZ-5500，マルチ画面マウスサポート，グラフィックのきめ細かさは言い表わせない。ぜひ実物も！　前後してX1のCDモデルの発表。ポケットに入れても良し，郵便で送ってもよし，5″ディスクよりも丈夫なケース入り。CP/Mも発売されるとのこと。これからのパソコンはコンパクトディスク。(H)

▶最近まで日本ではなじみのうすかった，マウスなるものが進出しはじめてた。この方式は光学式とボールを入れたものの2種類がある。現在では入手可能で，MZ-5500シリーズには，前者が採用されている。元々コンピュータと人間の操作性の向上のために生まれたものであるが，小生は，足で使えるマウスがあれば，両手も使えていいのになぁと思っている。誰か作って！　(M)

▶データショウ'83を前にして，シャープからは数多くのマシンが発売されました。16bitのMZ-5500は，黒色ながらかわいらしいマウスが付き，X1シリーズには，マイクロフロッピーが装備されました。ところでX1やMZ-2000シリーズ用のマウスが発売されないでしょうか。マウスにも兄弟がいたほうが……　(O)

▶台風のためか，ここ数日外は雨・雨・あめ。いいかげんうんざりしてきた。今日もまた雨かなと思いきや，カラッと晴れて雨あがりのたいへんすがすがしい朝を迎えた。こんなに良い天気なのに涼しいというよりむしろ，多少寒い感じがします。もう，一雨ごとに秋に近づいているのですね。　(S)

▶暑い夏が去り，すずしい秋です。この季節，マイコン業界は新しい年に向けて，新製品の発表が相い次ぎます。とくに，この本が書店に並ぶ頃，データショウ'83が東京晴海で開かれます。シャープは今月号で紹介しているとおり，もうMZ-5500，X1と発表を済ましており多分このふたつが目玉かと思います。さて，他社はどうかな。　(GTG)

▶人々が寝静まった真夜中に，まったく物音のしない密室で耳をふさぐと「プチッ，プチッ」とかすかな音が聞こえます。これは脳細胞がひとつずつ破壊されていく音なのであると某有名作家が真剣に語っていました。あなたもぜひ体験してみてください。秋は大急ぎで通り過ぎ，あっという間に冬の気配を感じる今日このごろです。(A)

▶私にとってパソコンは，ひとつの驚異。つい最近までインベーダーゲームすらしたことがなかった人，それが私です。しかし，このごろは仕事に暇をみつけては，ひたすらLOGOで遊んでおります。ところで，唐突ですが，憂歌団は移籍する以前のほうが良かったと思いません？この辺を一度じっくりと考えたいと思うのですが…(R)

▶ひさしぶりにネクタイなどをしてくると（下はジーンズだが），みんなから，「どうしたのー」とひやかされます。だから，「今日はデートなのサ！」と言うことにしています。先日，何人かがネクタイをしめて来たので，どうしたのか聞くと，「今日は仕事なんだ！」と言われました。なにかがくるっているような……。　(T)

▶編集に新人が入りました。昔，後記を個人攻撃に悪用しているとして去った人がいましたが，ここらでまた，え～新人の三武るみさんは……。あ，スペースがない(K)

STRATFORD SOFT

監修 家庭教師センター 学習館 **C.A.I.〔教育用ソフト317種完成！〕** 全種取扱説明書付

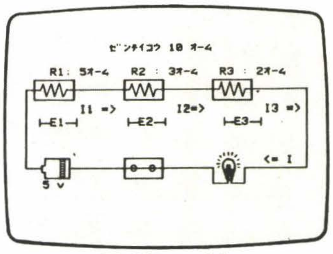
パソコン家庭教師

ストラットフォードの教育用ソフトは

- ■「家庭教師センター学習館」の監修のもとに作成されます。
- ■文部省の学習指導要領に準拠した内容を備え、しかも、楽しく学べるゲーム形式を採用しています。
- ■予習・復習に最適であるのは勿論、学習塾でも好評をいただいております。
- ■全種取扱説明書付。317種が完成しました。

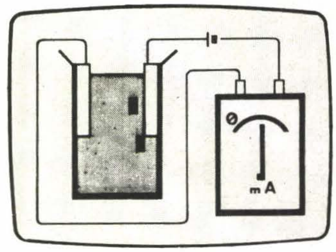
(実験シミュレーションシリーズは、すべてブック型パッケージ入りです。)

物理実験シミュレーション
ブック型 ¥4,800



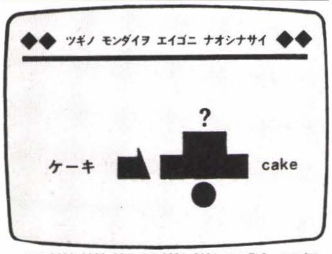
PC-8801, FM-7・8

化学実験シミュレーション
ブック型 ¥4,800



PC-8801・8001, FM-7・8, マルチ8

中学必修英単語
中1〜中3 各教科書別 ¥3,000



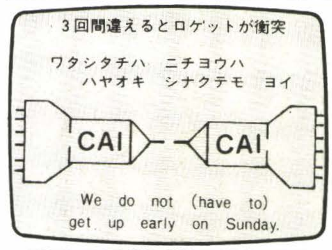
MZ-2200・2000・80B, PC-8001・6001, FM-7・8, パソピア, PHC-25, レベルIII, FP-1100・1000, X-07, マルチ8

中学必修英文法
中1〜中3 ¥3,300



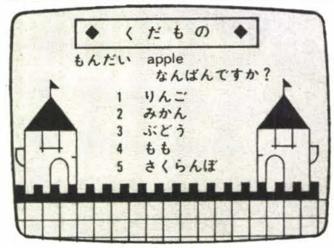
MZシリーズ, PC-8001・6001, FM-7・8, パソピア, PHC-25, JR-200, レベルIII, FP-1100・1000, X-07, マルチ8

中学必修英作文
中1〜中3 ¥3,300



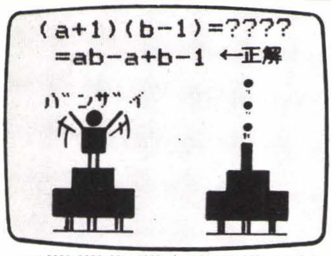
MZシリーズ, PC-8001・6001, FM-7・8, パソピア, PHC-25, JR-200, FP-1100・1000, マルチ8

幼児のえいご
3才以上 ¥3,300



PC-6001, JR-200, レベルIII, PHC-25, マルチ8

中学必修数学
中1〜中3 各上巻発売中 ¥3,300



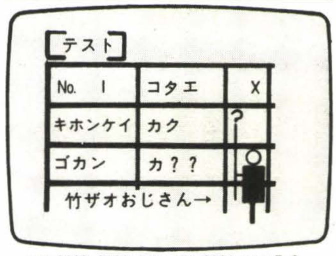
MZ-2200・2000・80B・1200K/C・K2E, PC-8001, FM-7・8, レベルIII, FP-1100・1000, JR-200, マルチ8

幼児教育〔かずあそび1〕
3才以上 ¥3,300



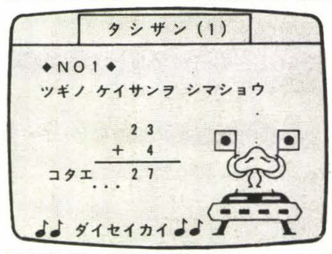
PC-6001, JR-200, レベルIII, PHC-25

中学必修国語〔文法〕
中学生向き ¥3,300



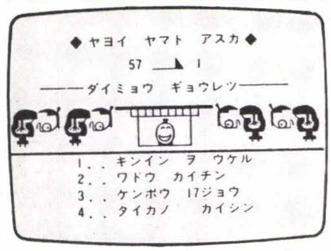
MZ-2200・2000・80B, PC-8001, FM-7・8, レベルIII, マルチ8

楽しい算数
小1〜小5 各上巻発売中 ¥3,300



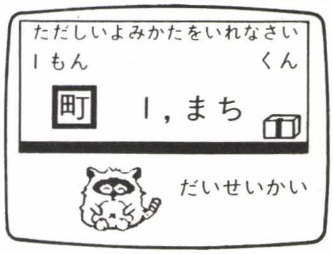
MZ-2200・2000・80B・1200, PC-8001・6001, JR-200, マルチ8, FP-1100・1000

日本史年表
中学生向き ¥3,300



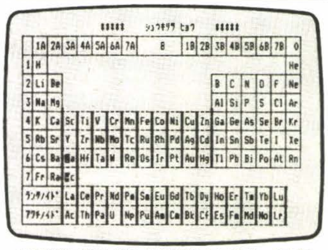
MZ-2200・2000・80B, PC-8001, FM-7・8, パソピア, JR-200, FP-1100・1000, レベルIII, マルチ8

楽しい国語〔漢字〕
小1・小2 要V-RAM1 ¥3,800



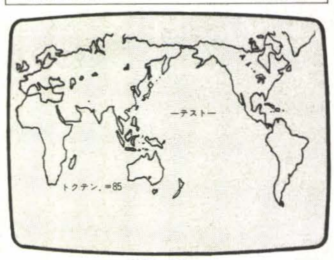
MZ-2200・2000・80B

化学〔元素記号マスター〕
中学生・高校生向 ¥3,300



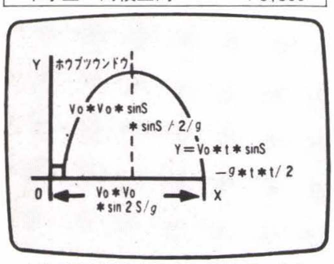
MZ-2200・2000・80B, PC-8001, FM-7・8, パソピア, レベルIII, X-07, FP-1100・1000, マルチ8, JR-200

世界地理〔河川・海流編〕
要V-RAM1 ¥3,800



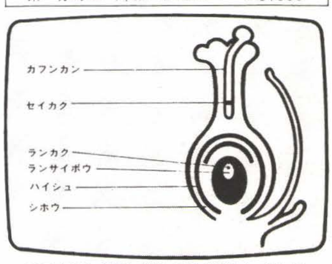
MZ-2200・2000・80B

物理〔運動と力〕〔エネルギー〕
中学生・高校生向 ¥3,300



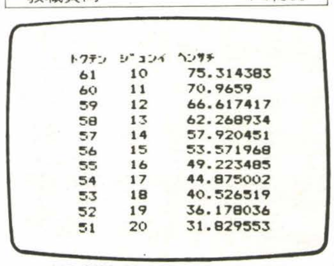
MZ-2200・2000・80B, PC-8001, FM-7・8, レベルIII

中学必修理科
第一分野上・下, 第二分野上・下 ¥3,300



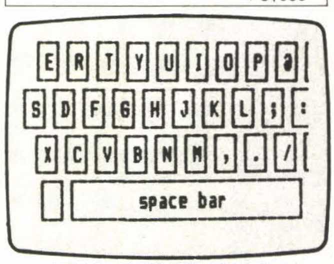
MZ-2200・2000・80B, PC-8001, レベルIII

クラス別成績管理
教職員向 ¥4,500



MZ-2000・80B, PC-8001, FM-7・8

キーボードトレーナー
¥3,000



MZシリーズ PC-8001, FM-7・8

★新情報化時代への提案★

新発売!! ウス キオリジナル商品

MZ−700用 Tape Version ワードプロセッサー　　　　　　　　　　　￥14,000

夏目瀬石　V.2.0

今まで通りのテープベースでカナ漢字変換機能付（漢字ROM使用せず）

他機能　○センターリング　○アンダーライン　○反転文字　○レイアウトマップ

○葉書印字機能付　○右づめ及び左づめ

尚旧バージョンお持ちの方は、差額￥2,000　荷造送料￥500と旧バージョンテープを添えてお申し込下さい。

プログラム名	内容	機能	定価
スパイってつらいのね	どじなスパイですね，命令書を探して再起できるか？	MZ2000 MZ2200	￥4,000
SHOCK HOPPER	（ショックホッパー）宇宙のバーナ星人とショックガンを使っての大激戦	MZ−700 MZ2200	￥3,500
DOWN BOMB	（ダウン・ボン）地底人（グルパー）を爆弾をつかってドンドンやっつけよう。	MZ−700 MZ2200 X1	￥3,500
‥	アドベンチャーロールプレイングゲーム	X1 MZ2200	￥4,000

ウス キオリジナル ゲーム ソフト

プログラム名	内容	定価	MZ 700	MZ80 K/C	MZ 80B	MZ2000 MZ2200	X1	PA 7	PC8001 MII	FM 7
ブラックジャック	トランプゲーム	￥3,500	○	○	○	○	○	○	○	○
四人麻雀	パソコン相手に4人マージャンはいかが!	￥3,000	○	○	○	○			○	○
レーダー大作戦	索敵して位置を確認そして適切な攻撃をして下さい。うっかりするとあなたが全滅するかも。	￥3,000				○	○		◎	◎
ルーレット	ご家庭でラスベカス気分	￥3,500	○		○	○	○	○	○	○
KENO（キノ）	BINGOとよく似たゲームで一攫千金(25,000倍)を狙えるゲームで、投票用紙を用意すれば大勢でも楽しめます。	￥3,500	○	○	○	○	○	◎	○	○
ゲームジャングラー	ジャングルの大蛇とのたたかいです。	￥3,000	○	○						
TREATorTRICK	悪魔と喰うか喰われるかの大激戦。	￥3,000	○	○			○			
セブンスタットポーカー	トランプゲームの最高峰コンピュータとのかけひき。	￥4,000	○	○	○	○	○	○	○	○
花札　オイチョカブ	美しい花札でお楽しみいただけます。	￥4,000					○			
花札　花合せ	日本古来のゲーム、グラフイックの表わす花札は本物と見間違いそう。	￥4,000					○			
TV将棋盤	棋譜の研究等に使えます。	￥3,000			○	○	○	◎	◎	◎
カラー四人麻雀	グラフイックRAMを使って本物らしく。	￥4,000				Ⓖ1·2·3	Ⓖ			
徳川家康	アドベンチャー、シュミレーション等のゲームを駆使し、キャラクターを成長させ、家康に天下を取らせることができるか、第1弾 少年編 第2弾 桶狭間決戦前夜	第1巻 ￥4,000	○		○	○	○	◎	○	○
		第2巻	○		○	○	○	◎	◎	◎

○ 発売中　◎ 開発中　Ⓖ1·2グラフィクRAM必要

SHARP MZ−5500用 ソフト 発売中!!

プログラム名	内容	定価
本格的四人麻雀〔雀鬼〕	あなたは勿論、コンピュータ側もチー、ポン、カンをする。又五種類の思考型体を持ち、コンピューター側の負けがこむと、思考の型体を変えるので、思わぬ落し穴も……	￥6,800
FANTASTIC FOREST（ファンタスティック．フォーレスト）	あなたは、この森にお父さんと狩りに来たのはよかったが、お父さんとはぐれてしまいました。右も左もわからないあなたは、無事ふもとの町までたどりつくことが出来るか？	￥6,800
PLAY LAS−VEGAS（プレイ ラスベガス）	ブラックジャック、バカラ、KENO、ルーレットの4種類のゲームがあり、ルールはLAS−VEGASルールに準拠。	￥6,800
花札（花鳥風月）	花札のオイチョカブと花合わせの2種類が入っています。	￥6,800
7 STUD PORKER	相手のブラフを見破り、どれだけ相手の手を読みきるかが勝負の鍵。	￥6,800

ウス キオリジナル開発商品

新発売!!	MZ−700 セイコー・GP80D ケーブル・アダプター			￥6,500
	セイコープリンターケーブル TK−55010 共通	GP550E用	1.シャープX1につないで漢字がうてます。	￥6,000
		開発中	2.グラフイックキャラクターROM	￥5,000
		GP700M用	1.シャープX1につないでカラーがうてます サンプルプログラム(スクリーンコピー)(テープバージョン)付	￥8,500
		開発中	2.グラフイックキャラクターROM	￥5,000

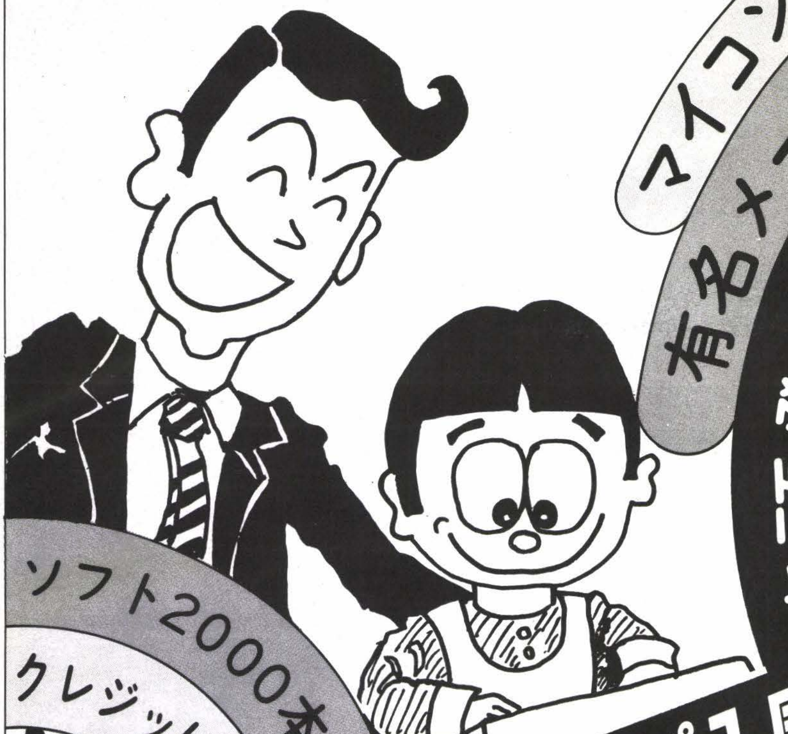
・MZ−700にセイコー・GP80Dを接続　・1行80文字　・MZ−700用・ワード・プロセッサー用に
MZ−700 漢字ワード・プロセッサー・漢字登録データー、ケーブルアダプターセット価格￥16,500

⊙荷作り送料は一律500円かかりますのでお願い申し上げます。⊙注文は現金書留で品名及び電話番号記入の上お願い申し上げます。
⊙注文の時はコンピューターの機械等くわしく記入の上お願い申し上げます。

株式会社 ウス キパソコンセンター

〒930 富山市総曲輪3の6の3☎(0764)21−4181(代)(内線25)
(0764)21−0897（直通）

●営業時間：午前10時➡午後5時
●定休日：毎週木曜日



マイコン有名メーカー 店頭発表価格奉仕中！

新製品情報
PC-6001mkII ¥84,800
MZ-2200 ¥128,000
OASYS2 セット¥480,000

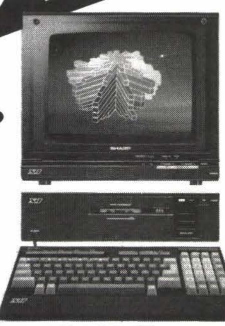
ソフト2000本
クレジットOK
チャンス！

マイコン・ショップ1周年セール中！

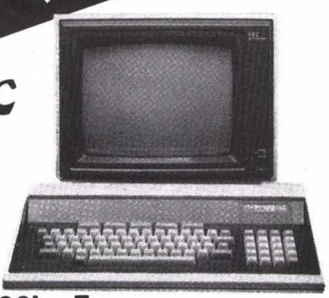
富士通



SHARP



NEC



PC-8001mkII ……………… ¥~~128,000~~
ベストセラー機PC－8001の使い易さをそのまま活かし、従来の豊富なソフトウェアをそっくり使えて、ビジネスからホビーまで活躍します。

PC－9801	¥~~298,000~~
PC－8801	¥~~228,000~~
PC－8201	¥~~138,000~~
PC－6001	¥~~89,800~~
PC－2001	¥~~59,800~~

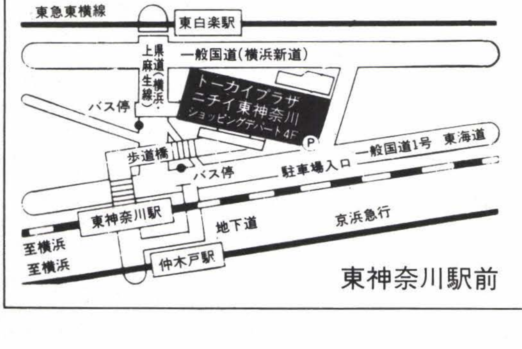
パソコンテレビX1
パーソナルコンピュータ＋キーボード
CZ-800C ¥~~155,000~~
14型カラーディスプレイテレビ………
CZ-800D ¥~~113,000~~
パソコンテレビ…………………………
CZ-800C/D ¥~~268,000~~
●コンピュータ・グラフィックス●ビデオ編集●コンピュータ学習●シンセサイザーミュージック●ワードプロセッシング●ホームコントロール●テレビゲーム●テレビコントロール●ビジネス計算

MZ－2000	¥~~218,000~~	PC－1251	¥~~29,800~~
MZ－700	¥~~79,800~~	PC－1250	¥~~22,800~~
PC－1500	¥~~59,800~~	PC－1245	¥~~17,800~~

FM-7（簡易言語ソフト付）
¥~~126,000~~
タモリでお馴みFM－7、簡易言語付ですから、購入したその日から、幅広い用途に活用できます。

FM－11EX(簡易言語付)……	¥~~398,000~~
FM－11AD(簡易言語付)……	¥~~338,000~~
FM－11ST………………………	¥~~268,000~~

カシオ	エプソン
FP シリーズ 3割引	フロッピーディスク プリンタ各種
	その他多数



東神奈川駅前

■お支払いは便利なクレジットで ボーナス一括払い実施中！

●お支払い例（¥215,000お買上げの場合）
国内信販でご利用の場合
月々わずか ¥23,650×10回
月々わずか ¥12,680×20回
●住友カード●ダイヤモンド●ユニオン●日本信販カード●ミリオンカード●日本ダイナーズ●JCB…………取扱店

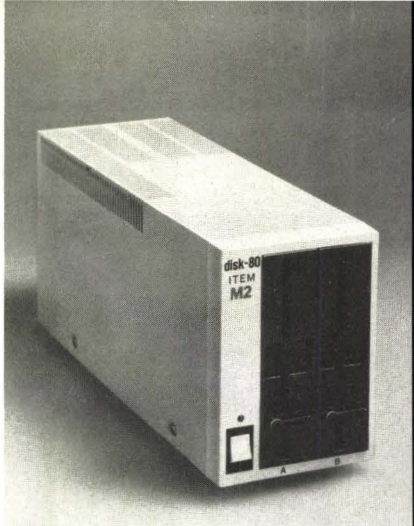
ペロ トーカイプラザ
ニチイと70の専門店・4F

横浜市神奈川区富家町1番地（東神奈川駅前）TEL.045(421)3005

5"デュアルディスクユニット

快走!disk-80markII

大好評のdisk-80markIIシリーズ。
高速に加え、さらに静寂・高性能。独走中。



FM-7/FM-8専用
disk-80FII　128,000円

■DISK BASICをはじめFLEXなど各DOSに対応。
■FM-7の高速バージョンの対応も可能です。■接続には disk-I/O F(15,000円)とCBL-3(5,000円)をご利用下さい。

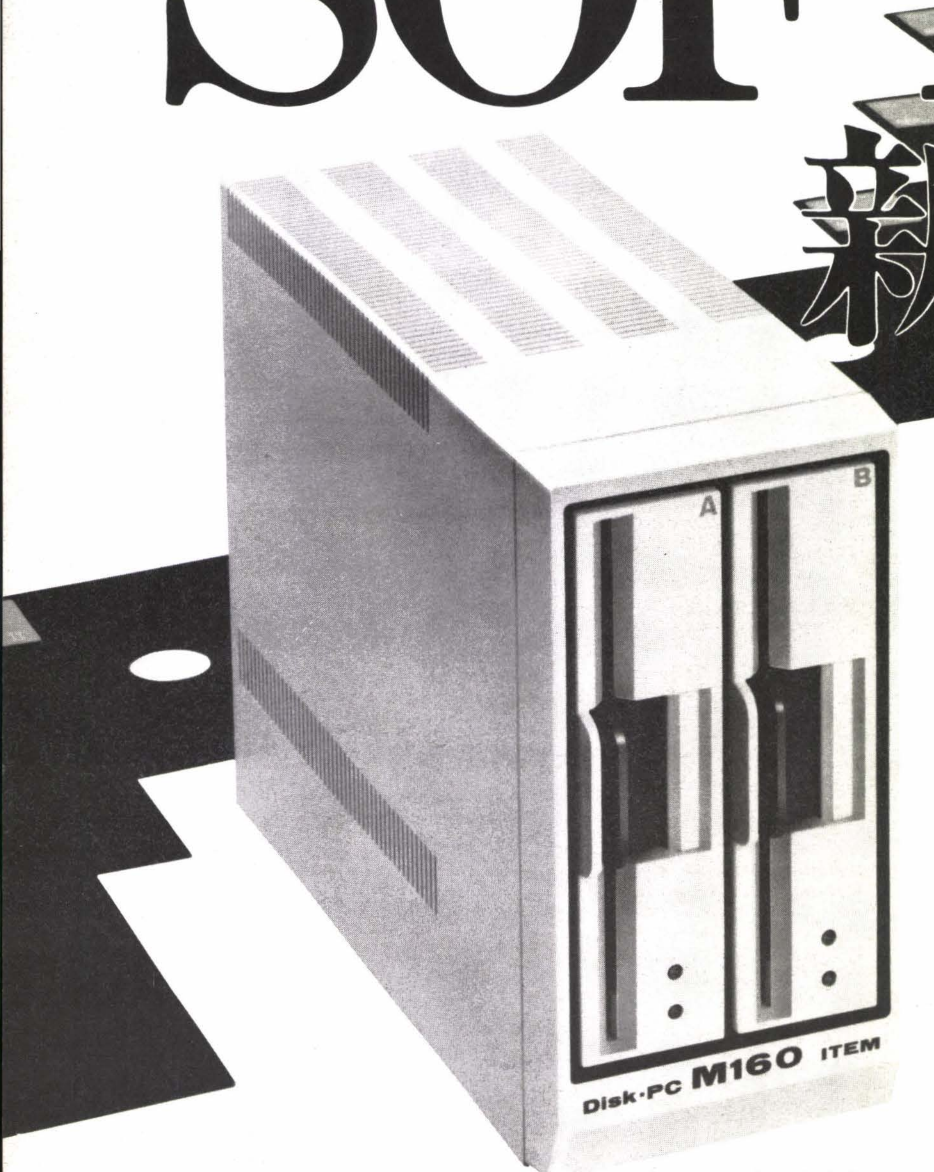
PC8801/PC9801/PC8001markII/PC8001専用
disk-80PII　123,000円

■RACET NEC DOS、CP/M対応 ■PC8001でご利用の場合はdisk-I/O P(12,000円)と接続ケーブルCBL-1(5,000円)をご利用下さい。他はCBL-2(5,000円)でダイレクト接続。システムディスケットは各対応のものをご利用下さい。

MZ-80B/MZ-2000専用
disk-80BII　138,000円

■フロッピーインターフェイス内蔵。拡張I/Oポートがあればあとは接続ケーブル(CBL-3)でダイレクトIN。

SOFT 新時代



高性能パワフル8インチフロッピーディスクユニット

Disk-PC M160

新発売 標準タイプ **256,000円**

CP/M、MS DOS、P-SYSTEM…………
数々のパワフルなSOFTが続々登場してきます。まさにSOFT新時代。

スリムな薄型設計。大容量1MB×2ドライブ。
ビジネスに、パーソナルデータベースに、
ハイコストパァーフォーマンスを提供いたします。

モデル

M160-01(PC9801用)…………………256,000円
M160-02(PC9801用CP/M-86付)……298,000円
M160-03(PC8801用)…………………296,000円

※M160-02以外はシステムディスク(DISK BASIC等)が必要です。
接続ケーブル、インターフェイス等はふくまれています。

株式会社 アイテム
〒251 藤沢市南藤沢8-1-202
TEL (0466)27-1668(代)

□お求めは全国マイコンショップへ。
□電話によるご注文もうけたまわっております。
0466-27-1668(AM9:00〜PM8:00)
□詳しいお問合せは左記あてへお申込み下さい。
□disk専用クレジットが用意されています。

＊CP/M$_{R}$はデジタル・リサーチ社の登録商標です。＊UCSD Pascal$_{TM}$はカリフォルニア大学理事会の登録商標です。＊FLEX$_{R}$はTSC社の登録商標です。

世界に先駆け ツクモが贈る

ツクモオリジナルワイヤレスジョイスティック

フラッシャージョイ

¥14,800送料¥1,000

この度、ツクモ電機では、世界に先駆けジョイスティックの究極の姿とも言うべきコードレスジョイスティックを開発致しました。
“ジョイスティックがコードレスである必要があるのか”と言うご質問もあると思いますが、我々はとにかく新しい事に挑戦してみたかったのです。
今後共、常に創造性にチャレンジするツクモの活躍にご期待下さい。



受光部
寸法：106×62×29㎜

送光部
寸法：100×74×38㎜

■使用可能機種
○Ｎ　Ｅ　Ｃ：PC-6001MKII、PC-6001
○コモドール：VIC-1001、64
○シャープ：パソコンテレビ X-1
○東　　芝：パソピア7
○ナショナル：JR-200

■主な仕様
○使用半導体：7IC、7トランジスタ
　4ダイオード
○伝送媒体：赤外光
○到達距離：3m前後
○電　　源：送光部 単3×2本
　受光部 9V ACアダプター付

お問い合わせは…
☎03(251)7395 ツクモオリジナル商品開発部 酒井迄

秋なのにHAL研セール!!

負けるな!!MZ-700「グラフィック機能」

（グラフィック機能＋サウンド機能と掲載したものがありましたがサウンド機能はついていません。
たいへん皆様にご迷惑をおかけしたことをお詫び致します。）



HAL研究所製 PCG-700
¥29.800

MZ-700ファン待望のメカ「PCG-700」¥29,800で登場!!

- PCG-700は“Programable Character Generator”の略称でMZ-700が持っているキャラクターを自由に作り変える事が出来る画期的な周辺機器です。
- 最大256種類のキャラクターが定義出来ます。
- 再定義により、無数の文字が表現可能です。
- MZ-700のメモリーは使用しません。
- PCGのON/OFFは基板上のスイッチ又は、プログラムで行ないます。
- PCG-700は、バスタイプですのでCPUを取りはずす必要はありません。

今、入手のチャンス!! **PCG-700組み合わせシリーズ** PCG-700＋ツクモ特選ソフト3本＝¥37,800→特価 **¥29,800** 送料¥1,200

通信販売のお問い合わせは
03(251)9911へ

ご注文の際は、ご注文になりたい商品名をご記入の上、現金書留にてお願い致します。20万円以上は銀行振込みも取り扱っております。お問い合わせ下さい

富士銀行　神田支店
普通口座　No.894047

即決 ツクモ全国クレジットOK!!
- 現金特別価格で各種クレジットが利用できます。残金のみに金利がかかります。
- 60回払いまで出来ます。但、1回の支払い額は3,000円以上。
- その場で、お持ち帰りできるクレジットもあります。
- 印鑑、身分証明書(免許証など)が必要です。尚、学生の方及び、未成年の方は、ご両親の保証も必要です。
- 取扱いクレジットカード：日本信販、JCB、DC、UC、VISA、MC、CF
- 尚、クレジットカードご使用の場合は、手数料がかかります。

■ご注文商品の中、未着の品物がございました時は、通販受付電話で納品書No.をお知らせ下さい。尚、商品の詳しい取り扱い等は、各販売店までお問い合わせ下さい

■通信販売は　〒101 東京都千代田区神田郵便局私書箱135 九十九電機㈱ MZII係

九十九電機株式会社

ニューセンター店　〒101 東京都千代田区外神田1-16-10 ☎03(251)0986～8
名古屋店3F　〒460 名古屋市中区大須3-30-86 ラジオセンター・アメ横ビル内 ☎052(263)1681
5号店　〒101 東京都千代田区外神田3-1-14 ☎03(251)0531～2
7号店　〒101 東京都千代田区外神田1-9-9 ☎03(253)4199
札幌店　〒060 札幌市中央区南二条西3-15-1 さっしんビルB1 ☎011(241)2299

■定休日　東京各店は毎週木曜日と第3水曜日、名古屋店は毎週月曜日

Enjoy Computer!

発売記念特価セール中!
シャープMZ-5500シリーズ
シャープX-1・C/X-1・D

あなたのパソコンへ心をこめて

フロッピーディスク
プリンター
モニター

周辺機器特売!

フロッピーディスク

PC-8031-2W ………………………… ¥148,000
MZ-80BF (I/Oカード、ケーブル付) ………………………… ¥149,000
LFD-550シリーズ (I/Oポート、ケーブル付) …………………………
………………………… ¥118,500 (MZ用のみ¥102,500)
TRS-80model I 用ミニフロッピーディスク
〔NO.0 (DOSケーブル取説付) … ¥45,000　NO.1 … ¥36,000　NO.2 … ¥36,000
NO.3 … ¥36,000　拡張インターフェイス (16K.32K) … ¥49,000〕

プリンター

MZ-80P6 (80桁多機能ドットプリンタ/I/Oカード接続ケーブルサービス) …………
………………………… ¥138,000
PC-8023-C (ドットマトリックスプリンタ) ………………………… ¥85,800
タンディ9インチラインプリンター (ケーブル別) ……… ¥29,800
PC-8821 (漢字ROM PC-8821-02 サービス) ………… ¥179,000

モニター

12インチグリーンモニター (各機種使用可) ………… ¥15,000

この価格で全ての本体にグリーンモニターサービス中! MZ-2000は除く

PC-6001mkII ………………………… ¥80,000
PC-8001mkII ………………………… ¥116,000
PC-8801 (漢字ROM PC-8801-01サービス) ……… ¥200,000
PC-9801 (漢字ROM PC-9801-01サービス) ……… ¥285,000
三菱マルチ8 ………………………… ¥116,000
TRS-80ml (限定特価) ………………………… ¥78,800

FM-7 ………………………… ¥125,000
カシオFP1100 ………………………… ¥69,800
パソピア7 ………………………… ¥115,000
MZ-2200 ………………………… ¥118,000
MZ-2000 ………………………… ¥135,000
《MZ-1U01 (シャープ拡張I/Oポート) ………………… 定¥3,700→特価¥22,000》

《15回均等分割払いの例》

	特別価格	分割払手数料	第1回お払い額	2回目以降お払い額	分割お支払い合計額
PC-8001MK2	¥116,000	¥18,560	¥9,960	¥8,900 ×14	¥134,560
PC-6001MK2	¥80,000	¥12,800	¥7,400	¥6,100 ×14	¥92,800
PC-8801	¥200,000	¥32,000	¥16,400	¥15,400 ×14	¥232,000
PC-9801	¥285,000	¥45,600	¥22,600	¥22,000 ×14	¥330,600
MZ-2000	¥135,000	¥20,250	¥11,050	¥10,300 ×14	¥155,250
MZ-2200	¥118,000	¥18,880	¥9,480	¥9,100 ×14	¥136,880
MZ-80BF	¥149,000	¥23,840	¥11,840	¥11,500 ×14	¥172,840
PC-8031-2W	¥148,000	¥23,680	¥12,080	¥11,400 ×14	¥171,680
FP-1100	¥69,800	¥11,200	¥5,600	¥5,400 ×14	¥81,200
FM-7	¥125,000	¥20,000	¥10,600	¥9,600 ×14	¥145,000
パソピア-7	¥115,000	¥18,400	¥10,200	¥8,800 ×14	¥133,400
マルチ8	¥116,000	¥18,560	¥9,960	¥8,900 ×14	¥134,560
TRS-80MI	¥78,800	¥12,640	¥6,240	¥6,100 ×14	¥91,640
MZ-80P6	¥139,800	¥22,240	¥11,440	¥10,700 ×14	¥161,240
PC-8023C	¥85,800	¥14,240	¥8,040	¥6,800 ×14	¥103,240
PC-8821 (漢字ROMボード付)	¥179,000	¥28,640	¥14,440	¥13,800 ×14	¥207,640
LFD-550	¥118,500	¥19,040	¥9,240	¥9,200 ×14	¥138,040



お支払方法/均等払い、ボーナス併用払い、ボーナス一括払い、3回〜36回までの分割払い等ございます。毎月銀行からの口座振替とさせて頂きます。
お申し込み/電話又はハガキでご連絡下さい。12インチグリーンモニターは機種をご指定下さい。

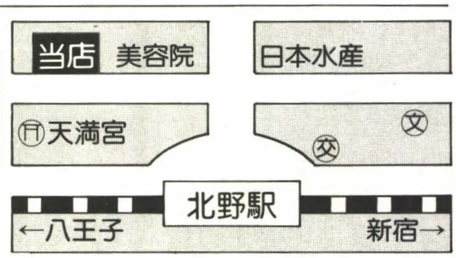
通信販売もご利用下さい。全商品、クレジットでも取扱います。

ご注文は在庫確認の上、現金書留又は、銀行振込みでお申込み下さい。

創業26年信用をモットーに信頼ある商品を安価にてお届け致します。

(有)アマノ電器サービスセンター

〒192 東京都八王子市北野町560-5
☎0426(45)3001(代)夜間可
*京王線北野駅下車徒歩3分
八王子よりバス北野駅行北野駅前下車　《水曜定休日　営業時間/AM10:00〜PM7:00》



※本誌発行時においてよりお求めやすい価格に変更されている場合がございますので価格を再度電話にてご確認下さい。

電話で深めようマイコンの世界

106 106 106 106 106 無料 106 106 106 106 106 無料 106 106 106 106 106

アコリット マイコン最新カタログ

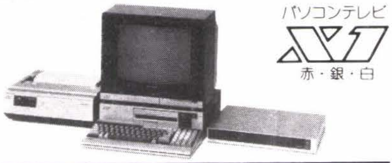
シャープ パソコンテレビX1

CZ-800C/D ¥ 268,000
48回分割 ¥ 6,400／60回分割 ¥ 5,400

CZ-800C ¥ 155,000
48回分割 ¥ 3,700／60回分割 ¥ 3,100

CZ-800D ¥ 113,000
48回分割 ¥ 2,700／60回分割 ¥ 2,300

型名	品名	定価/円	48分割	60分割
CZ- 8GR	グラフィックRAM	32,000→	800	600
CZ-8EP	拡張I/Oポート	11,800→	300	200
CZ-8KP	漢字ROM	38,000→	900	700
CZ-8FA	フロッピI/F	24,000→	600	500
CZ-800F	ミニフロッピディスク	198,000→	4,700	4,000
CZ-800P	ドットプリンタ	142,800→	3,400	2,900
CZ-8VC	RFビデオコンバータ	15,800→	400	300
CZ-8DT	デジタルテロッパー	89,800→	2,100	1,800
CZ-8R1	システムラック	29,800→	700	600
CZ-8R2	システムラック	29,800→	700	600



シャープ CRTシリーズ

型名	品名	定価/円	48分割	60分割
CZ-140DS	RGB14型カラー	89,800→	2,000	1,700
14M-131C	RGB14型カラー	69,800→	1,600	1,400
14M-142C	RGB14型カラー	99,800→	2 300	2,000
14M-132C	RGB14型カラー	118,000→	2,700	2,300
12M-212C	RGB12型カラー	99,800→	2,300	1,900
14M-111C	RGB14型カラー	67,800→	1,600	1,300
14M-112C	RGB14型カラー	118,000→	2,700	2,300
14M-114C	RGB14型カラー	168,000→	3,900	3,200
XM-140A	コンポジット14型C	74,800→	1,700	1,400
20M-202C	RGB20型カラー	175,000→	4,000	3,400
12M-15B	コンポジット12型G	29,800→	700	500
12M-13B	コンポジット12型G	39,800→	900	800
12M-18B	コンポジット12型G	44,800→	1,000	900

シャープ MZシリーズ

MZ-80B2 ¥ 278,000
48回分割 ¥ 6,600／60回分割 ¥ 5,600

MZ-2000 ¥ 218,000
48回分割 ¥ 5,200／60回分割 ¥ 4,400

MZ-1200 ¥ 148,000
48回分割 ¥ 3,500／60回分割 ¥ 3,000

シャープ ポケットコンピュータ

PC-1245 ¥ 17,800
48回分割 ¥ 400／60回分割 ¥ 300

PC-1500 ¥ 59,800
48回分割 ¥ 1,400／60回分割 ¥ 1,200

シャープ MZ-2200シリーズ

MZ-2200 ¥ 128,000
48回分割 ¥ 3,000／60回分割 ¥ 2,600

型名	品名	定価/円	48分割	60分割
MZ-1T02	データレコーダ	19,800→	500	400
MZ-1F07	ミニフロッピディスク	158,000→	3,800	3,200
MZ-1P07	80桁ドットプリンタ	79,800→	1,900	1,600
MZ-1M01	16ビットボードキット	78,000→	1,800	1,500
MZ-1R08	漢字ROMボード	29,000→	700	500
MZ-1D06	カラーディスプレイ	102,000→	2,400	2,000
MZ-1D09	14型TVモニタ	110,000→	2,600	2,200
MZ-80P4B	136桁ドットプリンタ	281,000→	6,700	5,700

SIC ¥187,000
48回分割 ¥5,300 ／ 60回分割 ¥4,500

SID ¥277,000
48回分割 ¥7,900 ／ 60回分割 ¥6,700

型名	品名	定価/円	48分割	60分割
MZ-1D15	カラーディスプレイ	72,000→	1,700	1,400
MZ-2Z001	ディスクBASIC	10,000→	200	200

シャープ MZ-3500シリーズ

MZ-3541 ¥ 410,000
48回分割 ¥ 9,800／60回分割 ¥ 8,300

MZ-3531 ¥ 320,000
48回分割 ¥ 7,600／60回分割 6,500

型名	品名	定価/円	48分割	60分割
MZ-1P8	漢字プリンタ	540,000→	12,900	10,900
CE-330P	136桁ドットプリンタ	159,000→	3,800	3,200
CE-333P	136桁ドットプリンタ	310,000→	7,400	5,900
MZ-6P01	I/Oカード	12,000→	300	200
MZ-1F02	ミニフロッピディスク	198,000→	4,700	4,000
MZ-1F03	増設フロッピディスク	90,000→	2,100	1,800
CE-331M	ミニフロッピディスク	350,000→	8,300	7,100
MZ-1F05	フロッピディスク	398,000→	9,500	8,100
MZ-1E03	I/Oカード	38,000→	900	700
MZ-1D02	グリーンディスプレイ	49,800→	1,200	900
MZ-1D03	カラーディスプレイ	163,000→	3,900	3,300
MZ-1R03	グラフィックボード	59,000→	1,400	1,100
MZ-1R04	グラフィックメモリ	12,000→	300	200
MZ-1R05	漢字メモリ	38,000→	900	700

注目のMSX対応各社パソコン

東芝"パソピアQ"HX-10D	¥65,800
(11月16日より) HX-10S	¥55,800
日立 MB-H1 (11月21日より)	¥62,800
三菱 ML-8000 (10月21日より)	¥59,800 (予定)
ソニー"HIT・BIT"HB-55 (10月21日より)	¥54,800
松下電器 CF2000 (10月末より)	¥54,800
三洋電機 MPC-10 (10月25日より)	¥74,800
日本楽器製造 YIS-303	未定
(発売日未定) YIS-503	〃
CX-5	〃
AX-501	〃

NEC 5200モデル05シリーズ

N5231 ¥ 648,000
48回分割 ¥15,400／60回分割 ¥13,100

N5231-02A ¥ 758,000
48回分割 ¥18,000／60回分割 ¥15,400

N5231-03A ¥ 548,000
48回分割 ¥13,000／60回分割 ¥11,100

型名	品名	定価/円	48分割	60分割
N5231-56	日本語付加機構	45,000→	1,000	900
N5231-57	ROMキット	25,000→	500	500

おもしろコンピュータ

アタリ ビデオコンピュータシステム2800
¥19,800→¥15,800

セガ コンピュータビデオゲーム
SG-1000 ¥15,000→¥12,000
SC-3000 ¥29,800→¥23,800

任天堂 ファミリーコンピュータ
¥14,800→¥11,800

お買得コーナー

■カシオ	EP-1100＋ソフト ＝¥70,000
■NEC	PC-8001(下取品) ＝¥50,000
■ナショナル	JR-100＋RFコンバーター ＋ソフト3巻＝¥33,000
■シャープ	MZ-700シリーズ MZ-711 →¥58,300 MZ-721 →¥65,600 MZ-731 →¥93,400

■オーディオ・キーボード・パーソナル無線もお問い合わせください。

チェンジアップコーナー

購入機種	下取機種	下取差額
NEC		
PC-6001MK II	PC-6001	¥55,000
PC-6001MK II	PC-8001	¥42,000
PC-6001MK II	MZ-731	¥46,000
PC-8001MK II	PC-8001	¥75,000
PC-8001MK II	PC-6001	¥81,000
PC-9801	PC-8801	¥160,000
PC-9801	PC-8801	¥217,000
富士通		
FM-7	PC-6000	¥91,000
FM-7	PC-8001	¥72,000
FM-11EX	PC-8801	¥270,000
FM-11EX	PC-8001	¥315,000
シャープ		
CZ-800C/D	PC-6001	¥178,000
CZ-800C/D	PC-6001 6042K	¥168,000
CZ-800C/D	PC-8001 12M-212C	¥168,000

●下取り価格は、ご使用期間及びご使用状態によって多少変わります。

大好評のビックリプレゼント継続中!!



全国どこからでも電話1本(無料)で、商品先取り！お支払いは楽々クレジットで胸キュン！です。

豊富なバリエーションの中から自由に組み合わせてください――

Acolyte
(アコリット)
〒110 東京都台東区北上野2丁目12番12号
営業時間／午前11時～午後7時 年中無休
☎03(844)3361

●掲載以外の商品もお問い合わせください。他にオーディオ、ビデオ、パーソナル無線も取り扱っています。クレジットは1～60回までOKです。
●初回金のみ、端数金額処理のため、上記金額よりも少し多くなります。

この他にも新製品があります。

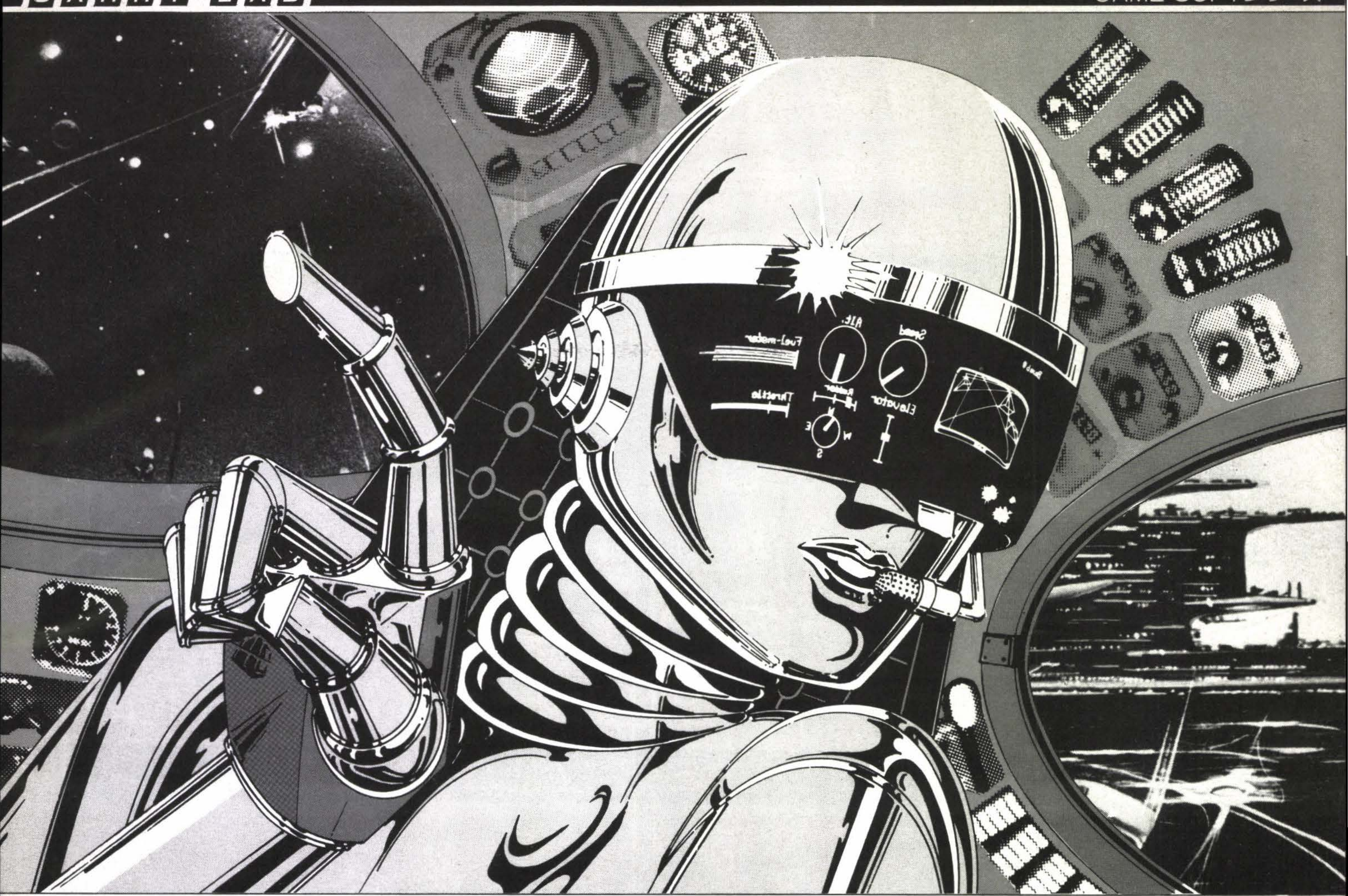
お支払いは楽な長期クレジット

48回/60回

●現金の場合はお問い合わせ下さい。クレジットは3,000円以上の組み合わせに願います。



CARRY LAB　　GAME SOFTシリーズ



超・未・来・空・間に挑め!!



FLIGHT SIMULATOR

PP 1
SHARP MZ-2200/2000用
GAME＝フライトシミュレーター&バクテリア
TAPE 2本／定価4,000円

PP 4
SHARP X1シリーズ
GAME＝フライトシミュレーター&スペースビー
TAPE 2本／定価4,000円

●**内容**：飛行機の操縦リアルタイム・シミュレーションゲーム。3次元から2次元透視変換によるリアルな画像を実現!! 一度、ご試乗ください。

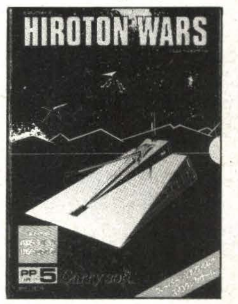
●**内容**：美しく流れる星々のもと、突如宇宙バチの軍団が出現した!! 激しく動きながら体当りしてくる宇宙バチに、君はどこまで耐えられるか!?

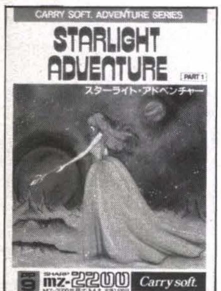
HIROTON WARS

PP 5
SHARP MZ-2200/2000用
GAME＝ヒロトンウォーズ
TAPE 1本／定価3,500円

PP 7
SHARP X1シリーズ
GAME＝ヒロトンウォーズ
TAPE 1本／定価3,500円

●**内容**：敵ミサイル、地雷をさけて敵戦車を破壊せよ!! 君はいつくのキャラクターを見ることができるか！チャレンジングステージが、君を待っている。





STAR LIGHT ADVENTURE

PP 9
SHARP MZ-2200/2000用
GAME＝スターライトアドベンチャー
TAPE 1本／定価3,500円

PP 10
SHARP X1シリーズ
GAME＝スターライトアドベンチャー
TAPE 1本／定価3,500円

●**内容**：俺はさまざまな宇宙の旅行者。今日は８つの星の太陽系にやってきた！　さあ何が起こるか？楽しみだぜ!!

ULTRA SEVEN BRIDGE

PP 13
SHARP MZ-2200/2000用
GAME＝ウルトラセブンブリッジ
TAPE 1本／定価3,500円

PP 14
SHARP X1シリーズ
GAME＝ウルトラセブンブリッジ
TAPE 1本／定価3,500円

●**内容**：コンピュータ3人を相手の本格的セブンブリッジ。オールマシン語のまさに、ウルトラセブンブリッジ。●グラフィック機能を活かした美しい画面です。●7ゲームのトータルで得点を競います。●制限時間はありません。



MZ-700用 (TAPE 1本/¥3,000)

- スターブラスト
- ラウンドファイター
- スクランブルジョー

MZ-2200/2000 (TAPE 1本/¥3,000)

- ラブアンドラブ
- キックオフ
- バグファイター

ニデコムキャリーのフロントライン及びスペースクルーザーは当社で開発したものです。

◎新しいパートナー
マイコンハウスSPS
マイクロキャビン

Carry soft. Carry lab. ●ソフト開発 ゼロソフト

(有)キャリーラボ
熊本市大江6丁目25-25 金子ビル1F
TEL.0963(63)0211
電話ファクシミリ0963(63)0235-G2・G3

●全国マイコン販売店で取り扱っています。通信販売/御注文は現金書留か郵便振替を御利用ください。尚、送料として300円加算してください。郵便振替口座/熊本18846●カタログおよび、マイコン情報誌(チャレンジャーズ)ご希望の方は、氏名、住所、年令、職業を明記の上、切手200円を添えてお申し込み下さい。●サポートコーナー 月～金曜14:00～17:00時まで電話にて受け付けます。TEL.0963-63-0047

HuBASIC™

堂々完成!

HuBASIC Version 2.0

HUDSONオリジナルBASIC Ver 2.0

MODEL:SHARP MZ-2000　MODE:IPL

Price:¥10,000　Serial WA-3600

カセット・マニュアル付



MZ-2000 HuBASIC Ver2.0の豊富な機能

1）200種を越える豊富な命令群

2）MZ-2000のカラー機能をフルサポート
タイリング・ペイント，点線，破線，WINDOW命令による強力なカラーグラフィック。

3）命令語の省略が可能(例 RUN→R., LIST→L.)

4）ファイル・ディスクリプタによる自在なファイル管理が可能
4Mバイトまでの範囲内で外部メモリーの設定が可能
㈱I･Oデータ機器発売中の大容量RAMボード（PIO-3034(320Kバイト))★を組み合わせることによって大量のデータ処理，プログラム処理が可能となります。
なお，グラフィックボード(MZ-1R01)を使用してのファイル，データ処理も可能です。

5）漢字出力機能
同じく㈱I･Oデータ機器発売中の漢字ROMボード★★を組み合わせることによってMZ-2000が簡単に漢字を出力することが出来ます。
カナ⇨漢字コード変換機能もサポートしています。

6）演算精度は整数，単精度8桁，倍精度16桁を持ち，かつ倍精度演算による関数計算もできます。

7）グラフィック・ウィンドウの座標系の拡張
±38乗以内の範囲で設定できます。

8）演算スピードをさらに高速化
演算スピードはVersion1.シリーズより高速化を計り，さらに各ステートメント自体の実行スピードもアップさせています。

9）ロングバリアブルネーム(Long Variable Name)の採用 240文字までの変数名が使え，かつアップバーや英小文字も変数に使えますので,REPEAT-UNTIL,WHILE～WENDなどの構造化プログラミングに適した命令や字下げを有効に使うことによってプログラムの保守性を向上させることができます。

10）強力なスクリーンエディタによるプログラム編集効率の向上、変数を保護しながら編集可能，さらにプログラムをチェーンしても変数は保護されます。

11）構造化プログラミング命令の採用
FOR～NEXT,REPEAT～UNTIL,WHILE～WEND,LABELの各命令を持っています。

12）パソコンTV X1, MZ-700, MZ-80B(HuBASIC ver2.0管理下)の各機種のソフトウェアを読むことができます。
(ただじ各機種特有の機能はサポートされません)。

13）キーファンクションの大幅な採用

14）デバッグ用トレース機能の装備

15）PRINT USING命令の装備

16）近日発売予定のディスクベーシックとは完全なコンパチビリティを保っています。

17）MZ-2000，HuBASIC ver2.0を使用して外部メモリーに倍精度数値データを格納した場合。
MEM:グラフィックメモリ（MZ-1R01にメモリーをフル実装48Kの場合）
5119ヶのデータを格納。
EMM:320KバイトRAMボード(PIO-3034)の場合
39935ヶのデータを格納できます。

```
10 SCREEN0,0:OPTIONSCREEN2:INIT"MEM:"
20 CLS:INPUT"VDIM NAME=",F$:INPUT"VDIM ｻｲﾀﾞｲ ﾖｳｿ ｽｳ=",MX:M=INT(MX/32)
30 OPEN"R",1,"MEM:"+F$:IFLOF(1)=0THENFORI=0TOM:PUT#1,I:NEXT:GOTO50
40 IFLOF(1) M THEN CONSOLE:CLOSE:ERROR10
50 GET#1,0:X=-1:CONSOLE3,22
60 DIM BF$(31):FORI=0TO31:FIELD#1,I*8ASAA$,8ASBF$(I):NEXT
70 X=X+1:INPUT"ﾋｷｽｳ ﾉ ｱﾀｲ(-1 ﾃﾞ END)=",X:IFX<0GOTO110 ELSEGOSUB120:'D#=VDIM(X)
80 PRINT"VDIM(";X;") ﾉ ｱﾀｲ=";D#:N#=D#
90 INPUT"ｶｷｶｴﾙ ﾃﾞｰﾀｰ",N#:IFD#<>N#THEN D#=N#:GOSUB130:'VDIM(X)=D#
100 PRINT:GOTO70
110 PUT#1,LOC(1):CLOSE:CONSOLE:END
120 GOSUB140:D#=CVD(BF$(G2)):RETURN
130 GOSUB140:BF$(G2)=MKD$(D#):RETURN
140 IFMX<X THEN CONSOLE:CLOSE:ERROR9
150 G1=INT(X/32):G2=X-G1*32:IF G1=LOC(1) THEN RETURN
160 PUT#1,LOC(1):GET#1,G1:RETURN
```

注1 ★ PIO-3034, MZ-2000/80B用, 320キロバイトRAMボード¥98,000
MZ-2000/80Bの拡張I/Oユニットに納まる, I/Oポート接続の320KB RAMボードです。Z-80AのI/O命令にて，アクセスできフロッピー，フロッピーの増設代替としても使用できます。
用途：画像データのバッファ／計測データのバッファ／FDD代替。

注2★★PIO-3055，MZ-2000/80B用，漢字ROMボード ¥35,000
MZ-2000/80Bの拡張I/Oユニットに納まる，漢字パターン書き込み済ボードJIS第一水準＋αの3700文字のほか，ユーザー作成の漢字，マーク等 500字を追加できる特徴を持ちます。
(16×16ドット漢字3700字＋2764用EPROMソケット付)

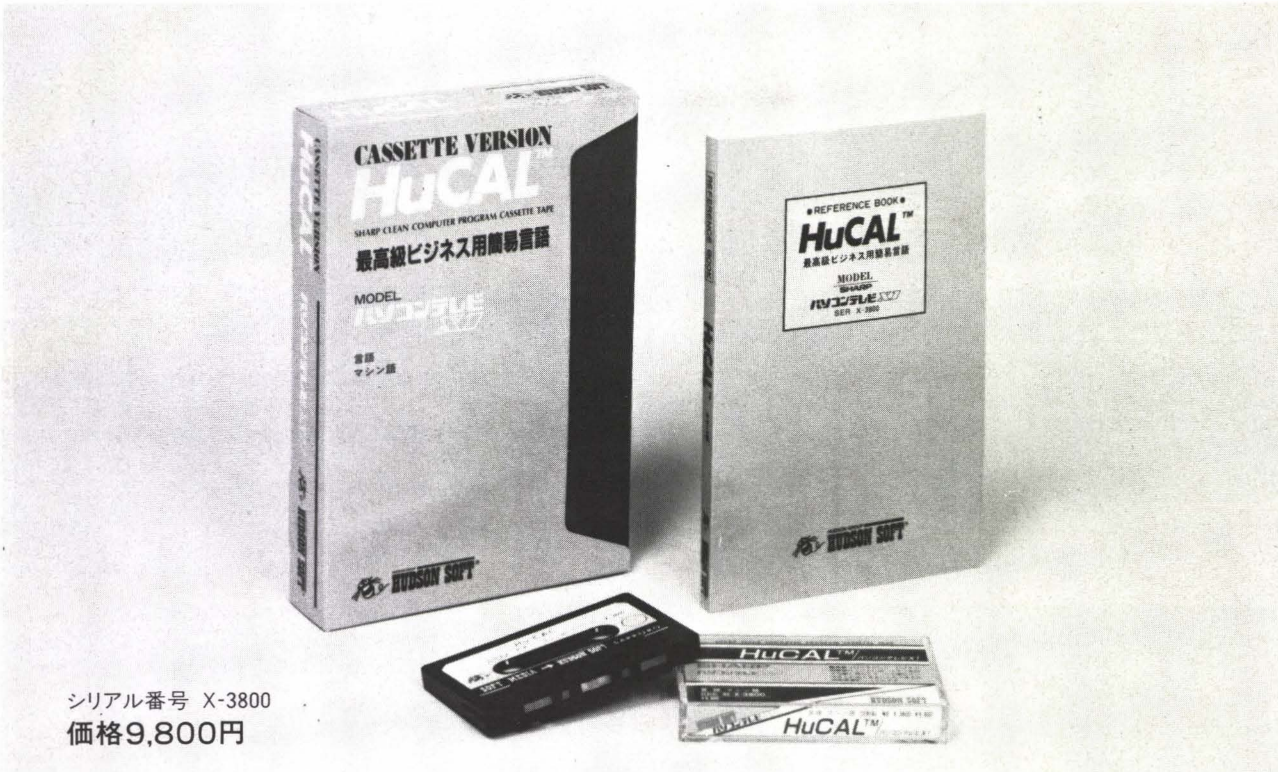
注1, 注2開発・販売元 ㈱I･Oデータ機器
〒920 石川県金沢市高岡町7-22　TEL.0762(21)4812

お問い合せは、最寄りのハドソン各営業所へ

ビジネス向け簡易言語

HuCAL™ テープ版

パソコンテレビX1用



■主な特長と概要

1. 16桁の高精度計算
2. 倍精度関数を装備
3. フルスクリーンエディタによる編集機能付
4. フォーマット指定が容易
5. フィルド調整はカーソルキーでOK
6. 縦列ごとに小数点以下の桁数指定可能
7. 縦列ごとに数字のカンマ付けの指定が可能
8. データのサーチ機能付
9. データのソート機能付
10. FOR~NEXTにより繰り返し処理が記述可能
11. @A%Lによりマクロ命令ジャンプとラベルジャンプの複合動作が可能
12. HDCP！（A，10）又はHDCP！（X，100）、230などの命令でハードコピーが可能
13. フィルドごとにデータの保護ができる
14. マクロ命令の文法エラーの場所がカーソルによって示される
15. ビジネスなどで特に使用する合計、平均は内部関数で簡単に使用できる
16. 表の大きさは横254×縦10000でメモリーの限界まで可能
17. マクロ命令のプリント出力がワンタッチでできる
18. INPUT ZAなどのマクロ命令において関数を含む式の入力ができる
19. MZ-80B/2000・MZ-80K/C/1200/700の各機種用Hucalのデータテープが読み書きできる
20. 倍精度関数を装備した事により、科学技術計算の集計用としてご利用できます
21. 簡単な計算は内部関数で処理し、複雑な計算はマクロ命令で計算します

TYPE II 新 発 売

MZ-2200	HuCAL II	WB-3801	¥12,800
MZ-80B	HuCAL II	W-3801	¥12,800
PC-8001mkII	HuCAL II	YB-3801	¥12,800
PC-8801	HuCAL II	YA-3801	¥12,800

＊上記の機種用タイプIIはX1用のものと同じ機能を有しています

HUDSON GROUP
HUDSON SOFT®
ハドソン札幌/〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸II201 PHONE:011-821-1538
ハドソン仙台/〒980 宮城県仙台市宮町1丁目4番28号 PHONE:0222-65-7031
ハドソン金沢/〒920 石川県金沢市本町2丁目1番28号 PHONE:0762-23-1263
ハドソン東京/〒102 東京都千代田区麹町4丁目7番5号 麹町ロイヤルビル2F PHONE:03-234-4996
ハドソン大阪/〒542 大阪市南区南船場4丁目2番18号 佐野屋橋ビル5F PHONE:06-251-1945
ハドソンUSA/2063 CENTER STREET BERKELEY CA 94704 TELEPHONE 415-845-1416

HuWP

日本語ワードプロセッサ



簡単に使え，豊富な機能

SHARP
パソコンテレビX1，MZ-2000,MZ-2200,MZ-80B

HuWPの概要

このワールドプロセッサーソフトはパソコン本体と純正プリンターに漢字ＲＯＭの組み合せで使用します。使用可能な文字は、漢字ＲＯＭに含まれているＪＩＳ第１水準の2965字及び、片かな、ひらがな、英数字、一部の記号などは直接キーボードから、又漢字はかな漢字変換方式で簡単に入力することができます。この漢字かな変換は従来のテープベースのものでは非常に困難でしたが、かな漢字変換ＲＯＭを使用することにより、本体のメモリー空間を使用することなく約6400字の読み方辞書を持つことができました。また、ＲＯＭベースですので変換は大変高速に行えますし、もちろん漢字パターンの表示も漢字ＲＯＭのおかげで高速に表示します。

文章の編集機能も強力でコントロールコードを用いることによって、文章のコピー、移動、インサート(挿入)、デリート(削除)などが自由自在に行えます。また、文字のサイズは倍角、全角、半角が自由に選べアンダーラインも引くことが出来ます。作成した文書はカセットテープに記録または再生することが出来、それに要する時間は１ページ当り１分以内と高速になっています。

以上のようにこのワードプロセッサソフトHuWPは、フロッピーディスクを使用しない物の中では豊富な機能を持ち、フロッピーを使用しないことにより、トータルコストも大変安価になっています。

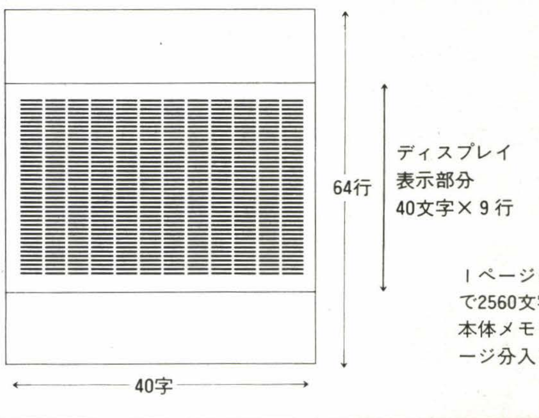
HuWPの特長

1）カナ漢字変換ＲＯＭの使用で漢字の入力が大変容易になった。
2）１ページには全角文字2,560文字を収容でき、本体メモリー内に２ページ記憶できる。
3）カセットベースのワードプロセッサでありながら、高機能である。
4）画面上に不要なマークを使用せず、印字される状態のまま表示になる。(倍角、半角なども画面上に表示される。)
5）用紙のサイズに合せて、色々な印字状態が設定できる。
6）文章全体のレイアウトを表示することができ、レイアウト表示中でも、通常の入力修正が出来る。
7）文章を１ページ、１分以内に記録再生できる。
8）カナ変換の時、読み方の前後をワンタッチで検索してくれる。
9）親切なHELP機能がある。
10フルマシン語でキータッチが良い。(X1は一部BASICで記述)

HuWPを使用する際必要なシステム

	パソコンTVX1	MZ-2200&MZ-2000
本　体	CZ-800C	MZ-2200 or MZ-2000
プリンター	CZ-800P	MZ-80P6 or 新MZ-80BP5 EPSON MP-80 TYPE3,100 TYPE3 RP-80, FP-80はPIO-3050typeMP06 を接続すれば使用可
漢字ROMボード	CZ-8KR	PIO-3055
グ ラ フィックRAM	CZ-8GR	MZ-1R01＋MZ-1R02(MZ-2000の場合) MZ-2200の場合は内蔵済

HuWPのページ構成



HuWPの実行

X1の場合

1）電源を入れ、本体附属のBASICテープをまずLOADして下さい。
2）BASICのLOAD終了後、本WPソフトをカセットにセットし、"RUN"して下さい。
3）マシン語部分を読み込みプログラムはAUTO STARTして初期画面になります。

MZ-2200&MZ-2000の場合

1）電源を入れ、本WPソフトをカセットにセットしてLOADさせて下さい。
なおBASICテープは必要ありません。
2）ソフトテープ読込み終了後、AUTOSTARTします。

MODEL：SHARP パソコンテレビX1	SERIAL X-3052-G ￥24,800（変換ROM付）
MODEL：SHARP MZ-2000, MZ-2200	SERIAL WB-3500A￥24,800（変換ROM付） WB-3500B￥19,800（変換ROM無）
MODEL：SHARP MZ-80B	SERIAL W-3500A ￥24,800（変換ROM付） W-3500B ￥19,800（変換ROM無）

HUDSON GROUP
HUDSON SOFT®
ハドソン札幌/〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸II201 PHONE:011-821-1538
ハドソン仙台/〒980 宮城県仙台市宮町1丁目4番28号 PHONE:0222-65-7031
ハドソン金沢/〒920 石川県金沢市本町2丁目1番28号 PHONE:0762-23-1263
ハドソン東京/〒102 東京都千代田区麹町4丁目7番5号 麹町ロイヤルビル2F PHONE:03-234-4996
ハドソン大阪/〒542 大阪市南区南船場4丁目2番18号 佐野屋橋ビル5F PHONE:06-251-1945
ハドソンUSA/2063 CENTER STREET BERKELEY CA 94704 TELEPHONE 415-845-1416



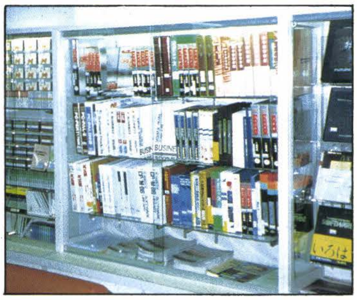
●J&Pはソフトとハードをシステムで提案いたします。

OA機器



OA機器コーナーでは他では見られない各メーカー有力機器が比較検討できます。
ワードプロセッサー・ファクシミリなどOA化には欠かせない機器を取揃えています。

ビジネスソフト



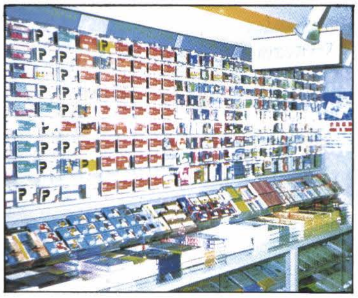
J&P自慢のソフトコーナーは豊富なアプリケーションソフトがギッシリ！目的に合ったソフトの選択から特別オーダーも承ります。

世界のパソコン



国内・海外の人気機種がズラリ勢揃い。あなたの目的に合った機種がご自由にお選びいただけます。
また、ディスプレイ、プリンタ、フロッピィディスクなど周辺機器を集中展示しております。

ホビーソフト



ビジネスソフトと合わせてJ&P自慢の大ソフトコーナーです。全国から集めた優秀なソフトは3,000種類。お求めやすく機種別に分けた展示が好評です。

パソコン教室(受講者募集)



初めての方でも、マニアの方にでも分かりやすく、楽しいカリキュラムの各講座をご用意いたしております。
希望のコースでお一人様一台の機種を使っていつでも受講いただけます。

●お問合わせ、お申し込みは下記のパソコン教室までお電話でどうぞ。

渋谷店教室	(03) 496-4148
テクノランド教室	(06) 644-5566
メディアランド教室	(06) 644-1613
京都寺町店教室	(075)341-3573
くずは店教室	(0720)56-8181
四日市店教室	(0593)54-3366
津店教室	(0592)26-0111



大阪日本橋 テクノランド 3F に
ビジネスソフトステーション誕生！

顧客管理、給与計算、在庫管理等あなたの会社の実情に即したビジネスソフトの選択について専門のアドバイザー達が、ご相談に応じます。
また特別オーダーソフトの設計も承りますので、お気軽にお問い合わせください。

Personal Computer Store

J&P

東京 渋谷店
東京都渋谷区道玄坂2丁目28番4号(〒150)
☎(03)496−4141

東京 新宿店
東京都新宿区新宿3丁目30番11号 高野第2ビルB1(〒160)
☎(03)352−3131

大阪日本橋 テクノランド
大阪市浪速区日本橋5丁目6番7号(〒556)
☎(06)644−1413

大阪日本橋 メディアランド
大阪市浪速区日本橋4丁目11番3号(〒556)
☎(06)644−1613

京都 京都寺町店
京都市下京区寺町通仏光寺下ル恵美須之町549(〒600)
☎(075)341−3571

大阪 くずは店
枚方市楠葉並木町2丁目2−2(〒573)
☎(0720)56−8181

大阪 千里中央店
豊中市新千里東町1-3-204 千里サンタウン3F (〒565)
☎(06)834−4141

名古屋 栄ノバ店
名古屋市中区栄3丁目4-5 SAKAE NOVA6F(〒460)
☎(052)261−9201

三重 四日市店
四日市市鵜の森2−1−19(〒510)
☎(0593)54−3366

三重 津店
津市丸之内31−20(〒514)
☎(0592)26−0111

Oh!MZ ㈱日本ソフトバンク発行 Printed in Japan 定価480円 雑誌02179―11